IDAMS

Pacote de Programas para a Análise e Gerenciamento de Dados Desenvolvido Internacionalmente

Manual de Referência do WinIDAMS

(lançamento 1.3)

Abril de 2008

Copyright © UNESCO 2001-2008

Publicado por
UNESCO, Organização das Nações Unidas
para a Educação, a Ciência e a Cultura
7, Place de Fontenoy
75352 Paris 07 SP, França

Título da obra original:
WinIDAMS Reference Manual (release 1.3)
© 2001-2008 by UNESCO
Primeira edição em inglês pela UNESCO em 1988

Tradução em português:
Prof. José Raimundo CARVALHO
Professor de Economia
CAEN Pós-graduação em Economia Universidade Federal do Ceará, Fortaleza, Brasil

ISBN 92-3-102577-5 (UNESCO - versão em inglês)

Prefácio

Objetivos do IDAMS

A idéia por trás do IDAMS é prover gratuitamente os Estados Membros da UNESCO de um pacote de programas de geranciamento de dados e análise estatística razoavelmente completo. O IDAMS, usado em combinação com o CDS/ISIS (software da UNESCO para administração de base de dados e recuperação de informação), os equipará com software integrado, permitindo o processamento, de uma maneira unificada, tanto de dados numéricos quanto textuais coletados com finalidades científica e administrativa por universidades, institutos de pesquisa, administrações nacionais, etc. O objetivo maior é ajudar os Estados Membros a progredir na racionalização da administração dos seus vários setores de atividades, um objetivo que é crucial tanto para estabelecer planos sólidos de desenvolvimento quanto para o monitoramento das suas execuções.

Origem e Breve História do IDAMS

IDAMS foi originalmente derivado do pacote de software OSIRIS III.2 desenvolvido no início dos anos setenta no Institute for Social Research da Michigan University, U.S.A. Ele tem sido, e continua sendo, enriquecido, modificado e atualizado pelo Secretariado da UNESCO com a cooperação de experts de diferentes países, como especialistas Americanos, Belgas, Britânicos, Colombianos, Franceses, Húngaros, Poloneses, Russos, Eslovacos e Ucranianos, de onde vem o nome do software:"Internationally Developed Data Analysis and Management Software Package".

No início havia IDAMS para computadores mainframe da IBM

O primeiro lançamento (1.2) foi distribuído em 1988; continha a maior parte das facilidades de análise de dados. Apesar de rotinas básicas e um número de programas terem sido retirados do OSIRIS III.2, eles foram substancialmente modificados e novos programas foram adicionados propiciando ferramentas para contagem de ordem parcial, análise fatorial, ordenamento de alternativas e tipologia com classificação ascendente. Inovações para manuseamento de códigos e para documentação de execução de programa foram incorporadas. O software foi acompanhado do Manual do Usuário, Amostras de Impressão e Cartão de Referência Rápida.

Lançamento 2.0 foi distribuído em 1990. Além de agrupar (1) programas para calcular correlações Pearsonianas e (2) programas para ordenamento de alternativas, ele continha melhorias técnicas em um número de programas.

Lançamento 3.0 foi distribuído em 1992. Continha melhorias significativas como: harmonização de parâmetros, palavras-chave e sintaxe de declarações de controle, possibilidade de checagem da sintaxe de declarações de controle sem execução, possibilidade de execução em um número limitado de casos, harmonização das menssagens de erro, possibilidade de agregação e listagem de variáveis recodificadas; recodificação alfabética e seis novas funções aritméticas na opção Recode. Dois novos programas foram adicionados: um para checar a consistência de dados e outro para análise discriminante. O Anexo com fórmulas estatísticas foi adicionado ao Manual do Usuário.

Nota: Em 1993, depois da preparação do lançamento 3.02 para os sistemas operacionais OS e VM/CMS, o desenvolvimento da versão para mainframe foi terminada.

Paralelamente, havia IDAMS para microcomputadores em MS-DOS

O desenvolvimento da versão para microcomputadores começou em 1988 e continuou em paralelo ao desenvolvimento da versão para mainframe até o lançamento 3.

O primeiro lançamento (1.0) foi distribuído em 1989, com as mesmas características e programas da versão mainframe.

Lançamento 2.0 foi distribuído em 1990; ele era também completamente compatível com a versão mainframe. Além disso, a Interface do Usuário propiciou facilidades para a preparação de dicionário, entrada de dados, preparação e execução de arquivos de setup e impressão de resultados.

Lançamento 3.0 foi distribuído em 1992 juntamente com a versão mainframe. Contudo, a Interface do Usuário se tornou bem mais amigável, propiciando novos editores de dicionário de dados, um acesso direto aos protótipos de setup para todos os programas, bem como um módulo para exploração de gráfica interativa de dados.

Os dois lançamentos intermediários 3.02 e 3.04, distribuídos em 1993 e 1994 respectivamente, incluíram principalmente melhorias técnicas internas e debugging de um número de programas. O lançamento 3.02 foi o último totalmente compatível com a versão mainframe.

Micro IDAMS iniciou sua existência independente em 1993. O software passou por um completo e sistemático processo de teste, especialmente na área de manuseio de erros do usuário, e foi completamente removido de bugs.

Lançamento 4 (último lançamento para DOS), distribuído em 1996, inclui uma interface amigável melhorada, possibilidade de adequação do ambiente, Manual do Usuário on-line, linguagem de controle simplificada, novas modalidades de apresentação gráfica e capacidade de produzir versões em línguas nacionais. Dois novos programas se originaram para dar aos usuários análise de agrupamentos e técnicas de busca de estruturas. O Manual do Usuário foi reestruturado para apresentar os tópicos de uma maneira concisa e fácil de seguir. Pela primeira vez, foi disponibilizada uma versão em Inglês.

Desde 1998, o lançamento 4 tem sido gradualmente desenvolvido em Francês, Espanhol, Árabe e Russo.

2000: primeira versão de IDAMS para Windows e desenvolvimento adicional

O lançamento 1.0 de IDAMS para o sistema operacional Windows de 32-bit foi fornecido para teste no ano 2000 e sua distribuição iniciou-se em 2001. Ela oferece uma interface de usuário moderna com uma gama de novas características para melhorar a facilidade de uso e o acesso on-line ao Manual de Referência usando Windows Help padrão. Novos componentes interativos para análise de dados propiciam ferramentas para construção de tabelas multidimensionais, exploração gráfica de dados e análise de séries temporais.

O lançamento 1.1 foi distribuído em setembro de 2002 com as seguintes melhorias: (1) externalização do texto, abrindo a possibilidade de ter o software em outras línguas além do Inglês; (2) harmonização do texto nos resultados. Foi a primeiro lançamento da versão Windows que apereceu em Inglês, Francês e Espanhol.

O lançamento 1.2 foi distribuído en julho de 2004 em Inglês, Francês e Espanhol, e contém novas funções em três programas, na Interface do Usuário e nos módulos interativos de exploração gráfica de dados e de análise de séries temporais. Foi distribuído en abril de 2006 em Português.

O lançamento 1.3 foi também distribuído em Inglês, Francês, Português e Espanhol, e contém um novo programa para análise de variância multivariada (MANOVA), cálculo do coeficiente de variação em quatro programas, manuseio melhorado das variáveis recodificadas com decimais em SCAT e TABLES, e completa harmonização do comprimento de registro de dados.

Reconhecimentos

Primeiramente, agradecimentos devem ir para o Prof. Frank-M. Andrews († 1994) do Institute for Social Research, University of Michigan, Estados Unidos, como também para o Instituto que autorizou a UNESCO a possuir o código fonte do OSIRIS III.2 e usá-lo como ponto de partida no desenvolvimento do pacote de software IDAMS. Maiores melhorias e adições vem acontencendo desde então. A esse respeito, agradecimentos particulares devem ir para: Dr Jean-Paul Aimetti, Administrador do D.H.E. Conseil, Paris e Professor no Conservatoire National des Arts et Métiers (CNAM), Paris (França); Prof. J.-P. Benzécri

and E.-R. Iagolnitzer, U.E.R. de Mathématiques, Université de Paris V (França); Eng. Tibor Diamant e Dr Zoltán Vas, József Attila University, Szeged (Hungria); Prof. Anne-Marie Dussaix, Ecole Supérieure des Sciences Economiques et Commerciales (ESSEC), Cergy-Pontoise (França); Dr Igor S. Enyukov and Eng. Nicolaï D. Vylegjanin, StatPoint, Moscow (Federação Russa); Dr Péter Hunya, que tem sido o Diretor do Kalmár Laboratory of Cybernetics, József Attila University, Szeged (Hungria), e Administrador do Programa IDAMS na UNESCO entre Julho de 1993 e Fevereiro de 2001; Jean Massol, EOLE, Paris (França); Prof. Anne Morin, Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA), Rennes (França); Judith Rattenbury, ex-Diretora, Divisão de Processamento de Dados, World Fertility Survey, London, e atualmente fundadora e diretora da SJ MUSIC editora, Cambridge (Reino Unido); J.M. Romeder e Association pour le Développement et la Diffusion de l'Analyse des Données (ADDAD), Paris (França); Prof. Peter J. Rousseeuw, Universitaire Instelling Antwerpen, (Bélgica); Dr A.V. Skofenko, Academy of Sciences, Kiev (Ucrânia); Eng. Neal Van Eck, Susquehanna University, Selinsgrove (Estados Unidos); Nicole Visart que lançou o Programa IDAMS na UNESCO e que, além das suas contribuições em todos os estágios, assegurou a coordenação e monitoramento de todo o projeto até a sua aposentadoria em 1992.

Éimpossível dar os devidos créditos a todas as pessoas, além dos acima já mencionados, que contribuíram com idéias e esforços para o IDAMS e para o OSIRIS III.2, de onde foi originado. Até agora IDAMS tem sido desenvolvido principalmente na UNESCO. Segue uma lista de nomes dos principais programas, componentes e facilidades incluídos no WinIDAMS, com os nomes dos autores e programadores, e os nomes das instituições onde foi realizado o trabalho.

Interface do Usuário e Facilidades Básicas

Recode	Ellen Grun	ISR
	Peter Solenberger	ISR
Interface do Usuário	Jean-Claude Dauphin	UNESCO
Acesso on-line ao	Pawel Hoser	Polish Academy of Sciences
Manual de Referência	Jean-Claude Dauphin	UNESCO

Facilidades para Gerenciamento de Dados

AGGREG	Tina Bixby	ISR
	Jean-Claude Dauphin	UNESCO
BUILD	Carl Bixby	ISR
	Sylvia Barge	ISR
	Tibor Diamant	UNESCO
CHECK	Tina Bixby	ISR
	Jean-Claude Dauphin	UNESCO
CONCHECK	Neal Van Eck	Van Eck Computing Consulting
CORRECT	Tibor Diamant	UNESCO
IMPEX	Péter Hunya	UNESCO
LIST	Marianne Stover	ISR
	Sylvia Barge	ISR
	Jean-Claude Dauphin	UNESCO
MERCHECK	Karen Jensen	ISR
	Sylvia Barge	ISR
	Zoltán Vas	JATE
MERGE	Tina Bixby	ISR
	Nancy Barkman	ISR
	Jean-Claude Dauphin	UNESCO
SORMER	Carol Cassidy	ISR
	Jean-Claude Dauphin	UNESCO
SUBSET	Judy Mattson	ISR
	Judith Rattenbury	ISR
	Jean-Claude Dauphin	UNESCO
TRANS	Jean-Claude Dauphin	UNESCO

Facilidades para Análise de Dados

CLUSFIND	Leonard Kaufman	Vrije Universiteit Brussel
	Peter J. Rousseeuw	Vrije Universiteit Brussel
	Neal Van Eck	Van Eck Computing Consulting
	Tibor Diamant	UNESCO
CONFIG	Herbert Weisberg	ISR
DISCRAN	J.-M. Romeder and ADDAD	ADDAD
	Péter Hunya	UNESCO
	Tibor Diamand	UNESCO
FACTOR	J.P. Benzécri,	Université de Paris V
	E.R. Iagolnitzer	Université de Paris V
	Péter Hunya	JATE
MANOVA	Charles E. Hall	George Washington University
	Elliot M. Cramer	George Washington University
	Neal Van Eck	ISR
	Tibor Diamand	UNESCO
MCA	Edwin Dean	ISR
	John Sonquist	ISR
	Tibor Diamant	UNESCO
MDSCAL	Joseph Kruskal	Bell Telephone
	Frank Carmone	Bell Telephone
	Lutz Erbring	ISR
ONEWAY	Spyros Magliveras	ISR
	Tibor Diamant	UNESCO
PEARSON	John Sonquist	ISR
	Spyros Magliveras	ISR
	Neal Van Eck	ISR
	Ronald Nuttal	Boston College
	Tibor Diamant	UNESCO
POSCOR	Péter Hunya	JATE
QUANTILE	Robert Messenger	ISR
	Tibor Diamant	UNESCO
RANK	Anne-Marie Dussaix	ESSEC
	Albert David	ESSEC
	Péter Hunya	JATE
	A.V. Skofenko	Ukrainian Academy of Sciences
REGRESSN	M.A. Efroymson	ESSO Corporation
	Bob Hsieh	ESSO Corporation
	Neal Van Eck	ISR
	Peter Solenberger	ISR
SCAT	Judith Goldberg	ISR
SEARCH	John Sonquist	ISR
	Elizabeth Lauch Baker	ISR
	James N. Morgan	ISR
	Neal Van Eck	Van Eck Computing Consulting
	Tibor Diamant	UNESCO
TABLES	Neal Van Eck	ISR and Van Eck Computing Consulting
	Tibor Diamant	UNESCO
TYPOL	Jean-Paul Aimetti	CFRO
	Jean Massol	CFRO
	Péter Hunya	JATE
	Jean-Claude Dauphin	UNESCO
Tabelas Multidimensionais	Jean-Claude Dauphin	UNESCO
GraphID	Igor S. Enyukov	StatPoint
	Nicolaï D. Vylegjanin	StatPoint
TimeSID	Igor S. Enyukov	StatPoint

Com relação à documentação, reconhecimentos devem ser expressados para todas as pessoas que contribuíram para a sua preparação, particularmente para: Judith Rattenbury que esboçou a primeira versão em Inglês do Manual (1988) e que continuou revisando outras edições até 1998; Jean-Paule Griset (UNESCO, Paris) que desenhou juntamente com Nicole Visart a tipografia do Manual usado até 1998; Teresa Krukowska (IDAMS Group, UNESCO, Paris) que compilou a parte com as fórmulas estatísticas, mudou a tipografia do Manual em 1998, continua atualizando a versão original em Inglês desde 1999, que é responsável pela produção do Manual em Inglês, Francês, Espanhol e Português, e que toma conta da harmonização, o máximo possível, dos textos em Inglês, Francês, Espanhol e Português.

Reconhecimento para os autores dos documentos do OSIRIS de cujo material foi extraído o Manual de Referência do WinIDAMS devem ser feitos da seguinte maneira: o Manual do Usuário do OSIRIS III.2 Vol.1 (editado por Sylvia Barge e Gregory A. Marks) e Vol.5 (compilado por Laura Klem), Institute for Social Research, University of Michigan, Estados Unidos.

Agradecimentos devem ir também para os tradutores do pacote de software e da documentação em Francês, Espanhol e Português pela suas cooperações:

- Professor José Raimundo Carvalho, CAEN Pós-graduação em Economia, UFC, Fortaleza, Brasil, pela tradução do Manual e dos textos pertencentes ao software em Português.
- Professor Bernardo Liévano, Escuela Colombiana de Ingeniería (ECI) Bogota, Colômbia, pela tradução do Manual e dos textos pertencentes ao software em Espanhol.
- Professor Anne Morin, Institut de Recherche en Informatique et Systèmes Aléatoires (IRISA), Rennes, França, pela contribuição na tradução em Francês dos textos pertencentes ao software.
- Nicole Visart, Grez-Doiceau, Bélgica, pela tradução do Manual em Francês.

As seguintes instituições realizaram traduções do software e do Manual em Árabe e Russo: ALECSO - Departamento de Documentação e Informação, Tunis, Tunisia, e Universidade Hidrometeorológica do Estado Russo, Departamento de Telecomunicações, St. Petersburg, Federação Russa.

Requisições de WinIDAMS and Maiores Informações

Para maiores informações do WinIDAMS relacionadas ao conteúdo, atualização, treinamento e distribuição, por favor escreva para:

UNESCO
Communication and Information Sector
Information Society Division
CI/INF - IDAMS
1, rue Miollis
75732 PARIS CEDEX 15
France
e-mail: idams@unesco.org
http://www.unesco.org/idams

Conteúdo

1 Introdução **1**
1.1 WinIDAMS Interface do Usuário . . . 1
1.2 Facilidades para Gerenciamento de Dados . . . 2
1.3 Facilidades para Análise de Dados . . . 2
1.4 Dados em IDAMS . . . 4
1.5 Comandos do IDAMS e Arquivo "Setup" . . . 5
1.6 Características Padrão do IDAMS . . . 5
1.7 Importação e Exportação de Dados . . . 6
1.8 Troca de Dados entre CDS/ISIS e IDAMS . . . 6
1.9 Estrutura deste Manual . . . 7

I Noções Fundamentais **9**

2 Dados em IDAMS **11**
2.1 O Dataset IDAMS . . . 11
2.1.1 Descrição Geral . . . 11
2.1.2 Método de Armazenamento e Acesso . . . 11
2.2 Arquivos Dados . . . 11
2.2.1 O arranjo de Dados . . . 11
2.2.2 Características do Arquivo Dados . . . 12
2.2.3 Arquivos Hierárquicos . . . 12
2.2.4 Variáveis . . . 12
2.2.5 Códigos para Dados Perdidos . . . 13
2.2.6 Valores Não-numéricos ou em Branco em Variáveis Numéricas - Dados Ruins . . . 13
2.2.7 Regras de Edição de Variáveis Produzidas pelos Programas do IDAMS . . . 13
2.3 O Dicionário do IDAMS . . . 14
2.3.1 Descrição Geral . . . 14
2.3.2 Exemplo de um Dicionário . . . 16
2.4 Matrizes IDAMS . . . 16
2.4.1 A Matriz Quadrada do IDAMS . . . 17
2.4.2 A Matriz Retangular do IDAMS . . . 18
2.5 Uso de Dados de outros Programas . . . 20
2.5.1 Dados Brutos . . . 20
2.5.2 Matrizes . . . 20

3 O Arquivo Setup do IDAMS **21**
3.1 Conteúdo e Finalidade . . . 21
3.2 Comandos do IDAMS . . . 21
3.3 Especificações de Arquivo . . . 23
3.4 Exemplos do Uso de Comandos $ e Especificações de Arquivo . . . 23
3.5 Declarações de Controle de Programa . . . 25
3.5.1 Descrição Geral . . . 25
3.5.2 Regras Gerais de Codificação . . . 25
3.5.3 Filtros . . . 25
3.5.4 Títulos . . . 27
3.5.5 Parâmetros . . . 27
3.6 Declarações de Recode . . . 31

4 Facilidade Recode **33**
4.1 Regras de Codificação . . . 33
4.2 Conjunto de Amostra de Declarações de Recode . . . 33
4.3 Manuseio de Dados Perdidos . . . 34
4.4 Como Recode Funciona . . . 34
4.5 Operandos Básicos . . . 35
4.6 Operadores Básicos . . . 36
4.7 Expressões . . . 36
4.8 Funções Aritméticas . . . 37
4.9 Funções Lógicas . . . 45
4.10 Declarações de Designação . . . 46
4.11 Declarações Especiais de Designação . . . 46
4.12 Declarações de Controle . . . 48
4.13 Declarações Condicionais . . . 49
4.14 Declarações de Inicialização/Definição . . . 50
4.15 Exemplos do Uso de Declarações de Recode . . . 52
4.16 Restrições . . . 54
4.17 Nota . . . 55

5 Gerenciamento e Análise de Dados **57**
5.1 Validação de Dados com IDAMS . . . 57
5.1.1 Resumo . . . 57
5.1.2 Checando a Integridade dos Dados . . . 57
5.1.3 Checando Valores Não-numéricas e Inválidas . . . 58
5.1.4 Checagem de Consistência . . . 59
5.2 Gerenciamento e Transformação de Dados . . . 59
5.3 Análise de Dados . . . 60
5.4 Exemplo de uma Pequena Tarefa a ser Executada pelo IDAMS . . . 60

II Trabalhando com o WinIDAMS **63**

6 Instalação **65**
6.1 Requerimentos do Sistema . . . 65
6.2 Procedimento de Instalação . . . 65
6.3 Testando a Instalação . . . 65
6.4 Folders e Arquivos Criados Durante Instalação . . . 66
6.4.1 Folders do WinIDAMS . . . 66
6.4.2 Arquivos Instalados . . . 66
6.5 Desinstalação . . . 67

7 Iniciando **69**
7.1 Visão Geral dos Passos a serem Executados com o WinIDAMS . . . 69
7.2 Criar um Ambiente de Aplicação . . . 70
7.3 Preparar um Dicionário . . . 71
7.4 Entre com Dados . . . 73
7.5 Prepare o Setup . . . 75
7.6 Execute o Setup . . . 76
7.7 Rever Resultados e Modificar o Setup . . . 77
7.8 Imprima os Resultados . . . 78

8 Arquivos e Folders **79**
8.1 Arquivos em WinIDAMS . . . 79
8.2 Folders em WinIDAMS . . . 80

9 Interface do Usuário **81**
9.1 Conceito Geral . . . 81
9.2 Menus Comuns a Todas Janelas do WinIDAMS . . . 82
9.3 Customização do Ambiente para uma Aplicação . . . 83
9.4 Criando/Renovando/Exibindo Arquivos Dicionário . . . 85

9.5 Criando/Renovando/Exibindo Arquivos Dados . . . 87
9.6 Importando Arquivos de Dados . . . 89
9.7 Exportando Arquivos Dados do IDAMS . . . 90
9.8 Criando/Renovando/Exibindo Arquivos Setup . . . 91
9.9 Executando Setups do IDAMS . . . 92
9.10 Manuseando Arquivos Resultados . . . 92
9.11 Criando/Renovando Arquivos em Formato Texto e RTF . . . 93

III Facilidades para Gerenciamento de Dados 95

10 Agregação de Dados (AGGREG) 97
10.1 Descrição Geral . . . 97
10.2 Características Padrão do IDAMS . . . 97
10.3 Resultados . . . 98
10.4 Dataset de Saída . . . 98
10.5 Dataset de Entrada . . . 99
10.6 Estrutura de Setup . . . 100
10.7 Declarações de Controle de Programa . . . 100
10.8 Restrições . . . 102
10.9 Exemplo . . . 102

11 Construção de um Dataset IDAMS (BUILD) 103
11.1 Descrição Geral . . . 103
11.2 Características Padrão do IDAMS . . . 104
11.3 Resultados . . . 104
11.4 Dataset de Saída . . . 105
11.5 Dicionário de Entrada . . . 105
11.6 Dados de Entrada . . . 105
11.7 Estrutura de Setup . . . 106
11.8 Declarações de Controle de Programa . . . 106
11.9 Exemplos . . . 107

12 Verificação de Códigos (CHECK) 109
12.1 Descrição Geral . . . 109
12.2 Características Padrão do IDAMS . . . 109
12.3 Resultados . . . 109
12.4 Dataset de Entrada . . . 110
12.5 Estrutura de Setup . . . 110
12.6 Declarações de Controle de Programa . . . 110
12.7 Restrições . . . 112
12.8 Exemplos . . . 112

13 Verificação de Consistência (CONCHECK) 115
13.1 Descrição Geral . . . 115
13.2 Características Padrão do IDAMS . . . 115
13.3 Resultados . . . 115
13.4 Dataset de Entrada . . . 116
13.5 Estrutura de Setup . . . 116
13.6 Declarações de Controle de Programa . . . 116
13.7 Restrições . . . 118
13.8 Exemplos . . . 118

14 Verificação de Intercalação de Registros (MERCHECK) 121
14.1 Descrição Geral . . . 121
14.2 Características Padrão do IDAMS . . . 123
14.3 Resultados . . . 123
14.4 Dados de Saída . . . 123
14.5 Dados de Entrada . . . 124
14.6 Estrutura de Setup . . . 124

14.7 Declarações de Controle de Programa . . . 124
14.8 Restrições . . . 127
14.9 Exemplos . . . 127

15 Correção de Dados (CORRECT) 129
15.1 Descrição Geral . . . 129
15.2 Características Padrão do IDAMS . . . 129
15.3 Resultados . . . 130
15.4 Dataset de Saída . . . 130
15.5 Dataset de Entrada . . . 130
15.6 Estrutura de Setup . . . 130
15.7 Declarações de Controle de Programa . . . 131
15.8 Restrição . . . 132
15.9 Exemplo . . . 132

16 Importação/Exportação de Dados (IMPEX) 135
16.1 Descrição Geral . . . 135
16.2 Características Padrão do IDAMS . . . 135
16.3 Resultados . . . 136
16.4 Arquivos de Saída . . . 136
16.5 Arquivos de Entrada . . . 137
16.6 Estrutura de Setup . . . 139
16.7 Declarações de Controle de Programa . . . 139
16.8 Restrições . . . 142
16.9 Exemplos . . . 142

17 Lista de Datasets (LIST) 145
17.1 Descrição Geral . . . 145
17.2 Características Padrão do IDAMS . . . 145
17.3 Resultados . . . 145
17.4 Dataset de Entrada . . . 146
17.5 Estrutura de Setup . . . 146
17.6 Declarações de Controle de Programa . . . 147
17.7 Restrição . . . 148
17.8 Exemplos . . . 148

18 Intercalação de Datasets (MERGE) 149
18.1 Descrição Geral . . . 149
18.2 Características Padrão do IDAMS . . . 149
18.3 Resultados . . . 150
18.4 Dataset de Saída . . . 150
18.5 Dataset de Entrada . . . 152
18.6 Estrutura de Setup . . . 152
18.7 Declarações de Controle de Programa . . . 153
18.8 Restrições . . . 155
18.9 Exemplos . . . 155

19 Classificação e Fusão de Arquivos (SORMER) 157
19.1 Descrição Geral . . . 157
19.2 Características Padrão do IDAMS . . . 157
19.3 Resultados . . . 157
19.4 Dicionário de Saída . . . 157
19.5 Dados de Saída . . . 157
19.6 Dicionário de Entrada . . . 158
19.7 Dados de Entrada . . . 158
19.8 Estrutura de Setup . . . 158
19.9 Declarações de Controle de Programa . . . 159
19.10Restrições . . . 159
19.11Exemplos . . . 160

20 Subdivisão de Datasets (SUBSET) 161
20.1 Descrição Geral . . . 161
20.2 Características Padrão do IDAMS . . . 161
20.3 Resultados . . . 161
20.4 Dataset de Saída . . . 162
20.5 Dataset de Entrada . . . 162
20.6 Estrutura de Setup . . . 162
20.7 Declarações de Controle de Programa . . . 163
20.8 Restrições . . . 164
20.9 Exemplos . . . 164

21 Transformação de Dados (TRANS) 165
21.1 Descrição Geral . . . 165
21.2 Características Padrão do IDAMS . . . 165
21.3 Resultados . . . 165
21.4 Dataset de Saída . . . 165
21.5 Dataset de Entrada . . . 166
21.6 Estrutura de Setup . . . 166
21.7 Declarações de Controle de Programa . . . 167
21.8 Restrições . . . 168
21.9 Exemplos . . . 168

IV Facilidades para Análise de Dados 171

22 Análise de Agrupamento (CLUSFIND) 173
22.1 Descrição Geral . . . 173
22.2 Características Padrão do IDAMS . . . 173
22.3 Resultados . . . 173
22.4 Dataset de Entrada . . . 174
22.5 Matriz de Entrada . . . 175
22.6 Estrutura de Setup . . . 175
22.7 Declarações de Controle de Programa . . . 175
22.8 Restrições . . . 177
22.9 Exemplos . . . 177

23 Análise de Configuração (CONFIG) 179
23.1 Descrição Geral . . . 179
23.2 Características Padrão do IDAMS . . . 179
23.3 Resultados . . . 179
23.4 Matriz de Configuração de Saída . . . 180
23.5 Matriz de Distância de Saída . . . 180
23.6 Matriz de Configuração de Entrada . . . 180
23.7 Estrutura de Setup . . . 181
23.8 Declarações de Controle de Programa . . . 181
23.9 Restrição . . . 182
23.10Exemplos . . . 183

24 Análise Discriminante (DISCRAN) 185
24.1 Descrição Geral . . . 185
24.2 Características Padrão do IDAMS . . . 185
24.3 Resultados . . . 185
24.4 Dataset de Saída . . . 186
24.5 Dataset de Entrada . . . 187
24.6 Estrutura de Setup . . . 187
24.7 Declarações de Controle de Programa . . . 188
24.8 Restrições . . . 190
24.9 Exemplos . . . 190

25 Funções de Distribuição e de Lorenz (QUANTILE) 191

25.1 Descrição Geral . . . 191
25.2 Características Padrão do IDAMS . . . 191
25.3 Resultados . . . 191
25.4 Dataset de Entrada . . . 192
25.5 Estrutura de Setup . . . 192
25.6 Declarações de Controle de Programa . . . 192
25.7 Restrições . . . 194
25.8 Exemplo . . . 194

26 Análise Fatorial (FACTOR) 195
26.1 Descrição Geral . . . 195
26.2 Características Padrão do IDAMS . . . 195
26.3 Resultados . . . 196
26.4 Dataset(s) de Saída . . . 197
26.5 Dataset de Entrada . . . 197
26.6 Estrutura de Setup . . . 198
26.7 Declarações de Controle de Programa . . . 198
26.8 Restrições . . . 201
26.9 Exemplos . . . 202

27 Regressão Linear (REGRESSN) 205
27.1 Descrição Geral . . . 205
27.2 Características Padrão do IDAMS . . . 206
27.3 Resultados . . . 207
27.4 Matriz de Correlação de Saída . . . 207
27.5 Dataset(s) de Resíduos de Saída . . . 208
27.6 Dataset de Entrada . . . 208
27.7 Matriz de Correlação de Entrada . . . 208
27.8 Estrutura de Setup . . . 209
27.9 Declarações de Controle de Programa . . . 209
27.10Restrições . . . 212
27.11Exemplos . . . 212

28 Escalonamento Multidimensional (MDSCAL) 215
28.1 Descrição Geral . . . 215
28.2 Características Padrão do IDAMS . . . 216
28.3 Resultados . . . 216
28.4 Matriz de Configuração de Saída . . . 217
28.5 Matriz de Dados de Entrada . . . 217
28.6 Matriz de Ponderação de Entrada . . . 218
28.7 Matriz de Configuração de Entrada . . . 218
28.8 Estrutura de Setup . . . 218
28.9 Declarações de Controle de Programa . . . 219
28.10Restrições . . . 220
28.11Exemplo . . . 221

29 Análise de Classificação Múltipla (MCA) 223
29.1 Descrição Geral . . . 223
29.2 Características Padrão do IDAMS . . . 224
29.3 Resultados . . . 225
29.4 Dataset(s) de Resíduos de Saída . . . 226
29.5 Dataset de Entrada . . . 226
29.6 Estrutura de Setup . . . 227
29.7 Declarações de Controle de Programa . . . 227
29.8 Restrições . . . 229
29.9 Exemplos . . . 230

30 Análise de Variância Multivariada (MANOVA) 233
30.1 Descrição Geral . . . 233
30.2 Características Padrão do IDAMS . . . 234

30.3 Resultados . . . 234
30.4 Dataset de Entrada . . . 235
30.5 Estrutura de Setup . . . 236
30.6 Declarações de Controle de Programa . . . 236
30.7 Restrições . . . 238
30.8 Exemplos . . . 238

31 Análise de Variância Univariada (ONEWAY) 241
31.1 Descrição Geral . . . 241
31.2 Características Padrão do IDAMS . . . 241
31.3 Resultados . . . 242
31.4 Dataset de Entrada . . . 242
31.5 Estrutura de Setup . . . 243
31.6 Declarações de Controle de Programa . . . 243
31.7 Restrições . . . 244
31.8 Exemplos . . . 245

32 Scoring Baseado em Ordenação Parcial de Casos (POSCOR) 247
32.1 Descrição Geral . . . 247
32.2 Características Padrão do IDAMS . . . 247
32.3 Resultados . . . 248
32.4 Dataset de Saída . . . 248
32.5 Dataset de Entrada . . . 248
32.6 Estrutura de Setup . . . 249
32.7 Declarações de Controle de Programa . . . 249
32.8 Restrições . . . 252
32.9 Exemplos . . . 252

33 Correlação de Pearson (PEARSON) 255
33.1 Descrição Geral . . . 255
33.2 Características Padrão do IDAMS . . . 255
33.3 Resultados . . . 256
33.4 Matrizes de Saída . . . 256
33.5 Dataset de Entrada . . . 257
33.6 Estrutura de Setup . . . 257
33.7 Declarações de Controle de Programa . . . 257
33.8 Restrições . . . 259
33.9 Exemplos . . . 259

34 Ordenamento de Alternativas (RANK) 261
34.1 Descrição Geral . . . 261
34.2 Características Padrão do IDAMS . . . 262
34.3 Resultados . . . 262
34.4 Dataset de Entrada . . . 263
34.5 Estrutura de Setup . . . 264
34.6 Declarações de Controle de Programa . . . 265
34.7 Restrições . . . 266
34.8 Exemplos . . . 267

35 Diagramas de Dispersão (SCAT) 269
35.1 Descrição Geral . . . 269
35.2 Características Padrão de IDAMS . . . 269
35.3 Resultados . . . 270
35.4 Dataset de Entrada . . . 270
35.5 Estrutura de Setup . . . 271
35.6 Declarações de Controle de Programa . . . 271
35.7 Restrições . . . 272
35.8 Exemplo . . . 272

36 Busca de Estrutura (SEARCH) 275

36.1 Descrição Geral . . . 275
36.2 Características Padrão do IDAMS . . . 275
36.3 Resultados . . . 276
36.4 Dataset de Resíduos de Saída . . . 276
36.5 Dataset de Entrada . . . 277
36.6 Estrutura de Setup . . . 277
36.7 Declarações de Controle de Programa . . . 277
36.8 Restrições . . . 280
36.9 Exemplos . . . 280

37 Tabelas Univariadas e Bivariadas (TABLES) 283
37.1 Descrição Geral . . . 283
37.2 Características Padrão do IDAMS . . . 284
37.3 Resultados . . . 284
37.4 Tabelas Univariadas/Bivariadas de Saída . . . 286
37.5 Matrizes de Estatísticas Bivariadas de Saída . . . 286
37.6 Dataset de Entrada . . . 286
37.7 Estrutura de Setup . . . 287
37.8 Declarações de Controle de Programa . . . 287
37.9 Restrições . . . 292
37.10Exemplo . . . 293

38 Tipologia e Classificação Ascendente (TYPOL) 295
38.1 Descrição Geral . . . 295
38.2 Características Padrão do IDAMS . . . 295
38.3 Resultados . . . 296
38.4 Dataset de Saída . . . 297
38.5 Matriz de Configuração de Saída . . . 297
38.6 Dataset de Entrada . . . 297
38.7 Matriz de Configuração de Entrada . . . 298
38.8 Estrutura de Setup . . . 298
38.9 Declarações de Controle de Programa . . . 298
38.10Restrições . . . 301
38.11Exemplos . . . 301

V Análise Interativa de Dados 303

39 Tabelas Multidimensionais e suas Apresentações Gráficas 305
39.1 Visão Geral . . . 305
39.2 Preparação da Análise . . . 305
39.3 Janela Tabelas Multidimensionais . . . 307
39.4 Apresentação Gráfica de Tabelas Univariadas/Bivariadas . . . 308
39.5 Como Fazer uma Tabela Multidimensional . . . 309
39.6 Como Mudar uma Tabela Multidimensional . . . 312

40 Exploração Gráfica de Dados 317
40.1 Visão Geral . . . 317
40.2 Preparação da Análise . . . 317
40.3 Janela Principal de GraphID para Análise de um Dataset . . . 317
40.3.1 Barra de Menu e Barra de Ferramentas . . . 318
40.3.2 Manipulação da Matriz de Diagramas de Dispersão . . . 320
40.3.3 Histogramas e Densidades . . . 322
40.3.4 Linhas de Regressão (Linhas suavizadas) . . . 322
40.3.5 Gráficos de Box-Whisker . . . 323
40.3.6 Diagrama Agrupado . . . 324
40.3.7 Diagramas de Dispersão em Três-dimensões e suas Rotações . . . 324
40.4 Janela de GraphID para Análise de uma Matriz . . . 325
40.4.1 Barra de Menu e Barra de Ferramentas . . . 326
40.4.2 Manipulação da Matriz Exibida . . . 327

41 Análise de Séries Temporais **329**
41.1 Visão Geral . . . 329
41.2 Preparação da Análise . . . 329
41.3 Janela Principal de TimeSID . . . 329
41.3.1 Barra de Menu e Barra de Ferramentas . . . 330
41.3.2 A Janela de Séries Temporais . . . 331
41.4 Transformação de Séries Temporais . . . 332
41.5 Análise de Séries Temporais . . . 333

VI Fórmulas Estatísticas e Referências Bibliográficas **335**

42 Análise de Agrupamento **337**
42.1 Estatísticas Univariadas . . . 337
42.2 Medidas Padronizadas . . . 337
42.3 Matriz de Dissimilaridade Computada de um Dataset do IDAMS . . . 338
42.4 Matriz de Dissimilaridade Computada de uma Matriz de Similaridade . . . 338
42.5 Matriz de Dissimilaridade Computada de uma Matriz de Correlação . . . 338
42.6 Partição ao Redor de Medoids (PAM) . . . 338
42.7 Agrupamento Aplicado a Grandes Volumenes de Dados (CLARA) . . . 340
42.8 Agrupamento Difuso (FANNY) . . . 340
42.9 Agrupamento Hierárquico Aglomerativo (AGNES) . . . 341
42.10Agrupamento Hierárquico Divisivo (DIANA) . . . 342
42.11Agrupamento Monotético (MONA) . . . 343
42.12Referências . . . 343

43 Análise de Configuração **345**
43.1 Configuração Centrada . . . 345
43.2 Configuração Normalizada . . . 345
43.3 Solução com Eixos Principais . . . 345
43.4 Matriz de Produtos Escalares . . . 346
43.5 Matriz de Distâncias Entre Pontos . . . 346
43.6 Configuração Rotacionada . . . 346
43.7 Configuração Translada . . . 346
43.8 Rotação Varimax . . . 346
43.9 Configuração Classificada . . . 347
43.10Referências . . . 347

44 Análise Discriminante **349**
44.1 Estatísticas Univariadas . . . 349
44.2 Discriminação Linear Entre 2 Grupos . . . 350
44.3 Discriminação Linear Entre Mais Do Que 2 Grupos . . . 351
44.4 Referências . . . 352

45 Funções de Distribuição e de Lorenz **353**
45.1 Fórmula para Pontos de Quebra . . . 353
45.2 Pontos de Quebra de Função de Distribuição . . . 353
45.3 Pontos de Quebra da Função de Lorenz . . . 354
45.4 Curva de Lorenz . . . 354
45.5 O coeficiente de Gini . . . 354
45.6 Estatística D de Kolmogorov-Smirnov . . . 354
45.7 Nota a Respeito de Pesos . . . 355

46 Análise Fatorial **357**
46.1 Estatísticas Univariadas . . . 357
46.2 Dados de Entrada . . . 358
46.3 Matrizes Núcleo (Matrizes de Relações) . . . 358
46.4 Traço . . . 359
46.5 Valores e Vetores Próprios . . . 359
46.6 Tabela de Valores Próprios . . . 360

46.7 Tabela de Fatores de Variáveis Ativas . . . 360
46.8 Tabela de Fatores de Variáveis Passivas . . . 362
46.9 Tabela de Fatores de Casos Ativos . . . 362
46.10Tabela de Fatores de Casos Passivos . . . 364
46.11Fatores Rotacionados . . . 364
46.12Referências . . . 364

47 Regressão Linear **365**
47.1 Estatísticas Univariadas . . . 365
47.2 Matriz de Soma Total de Quadrados e Produtos Cruzados . . . 365
47.3 Matriz de Soma de Quadrados Residuais e Produtos Cruzados . . . 366
47.4 Matriz de Correlação Total . . . 366
47.5 Matriz de Correlação Parcial . . . 366
47.6 Matriz Inversa . . . 366
47.7 Estatísticas de Resumo de Análise . . . 367
47.8 Estatísticas de Análise para Preditores . . . 368
47.9 Resíduos . . . 369
47.10Nota sobre Regressão Stepwise . . . 369
47.11Nota sobre Regressão Descendente . . . 370
47.12Nota sobre Regressão com Intercepto Zero . . . 370

48 Escalonamento Multidimensional **371**
48.1 Ordem de Computações . . . 371
48.2 Configuração Inicial . . . 371
48.3 Centragem e Normalização da Configuração . . . 371
48.4 História de Cálculos . . . 372
48.5 Esforço para Configuração Final . . . 374
48.6 Configuração Final . . . 374
48.7 Configuração Classificada . . . 374
48.8 Resumo . . . 374
48.9 Nota sobre Empates nos Dados de Entrada . . . 375
48.10Nota sobre Pesos . . . 375
48.11Referências . . . 376

49 Análise de Classificação Múltipla **377**
49.1 Estatísticas da Variável Dependente . . . 377
49.2 Estatísticas de Preditores para Análise de Classificação Múltipla . . . 378
49.3 Estatísticas de Análise para Análise de Classificação Múltipla . . . 379
49.4 Estatísticas de Resumo dos Resíduos . . . 380
49.5 Estatísticas de Categoria do Preditor para Análise de Variância Univariada . . . 380
49.6 Estatísticas para Análise de Variância Univariada . . . 381
49.7 Referências . . . 381

50 Análise de Variância Multivariada **383**
50.1 Estatísticas Gerais . . . 383
50.2 Cálculos para um Teste em uma Análise Multivariada . . . 385
50.3 Análise Univariada . . . 388
50.4 Análise de Covariância . . . 388

51 Análise de Variância Univariada **389**
51.1 Estatísticas Descritivas para Categorias da Variável de Controle . . . 389
51.2 Estatísticas de Análise de Variância . . . 390

52 Scoring Baseado em Ordenação Parcial de Casos **393**
52.1 Terminologia Especial e Definições . . . 393
52.2 Cálculo dos Escores . . . 394
52.3 Referências . . . 395

53 Correlação de Pearson **397**
53.1 Estatísticas Emparelhadas . . . 397
53.2 Médias e Desvios-Padrões Não-emparelhadas . . . 398

53.3 Equação de Regressão para Escores Brutos . . . 398
53.4 Matriz de Correlação . . . 398
53.5 Matriz de Produtos Cruzados . . . 398
53.6 Matriz de Covariância . . . 398

54 Ordenamento de Alternativas 401
54.1 Manuseamento dos Dados de Entrada . . . 401
54.2 Método Baseado em Lógica Clássica . . . 402
54.3 Métodos Baseados em Lógica Difusa: A Relação de Entrada . . . 404
54.4 Método difuso-1: Camadas Não-dominadas . . . 406
54.5 Método difuso-2: Ranges . . . 408
54.6 Referências . . . 409

55 Diagramas de Dispersão 411
55.1 Estatísticas Univariadas . . . 411
55.2 Estatísticas Univariadas Emparelhadas . . . 411
55.3 Estatísticas Bivariadas . . . 412

56 Busca de Estrutura 413
56.1 Análise de Médias . . . 413
56.2 Análise de Regressão . . . 415
56.3 Análise de Chi-quadrado . . . 416
56.4 Referências . . . 417

57 Tabelas Univariadas e Bivariadas 419
57.1 Estatísticas Univariadas . . . 419
57.2 Estatísticas Bivariadas . . . 420
57.3 Nota sobre Pesos . . . 426

58 Tipologia e Classificação Ascendente 427
58.1 Tipos de Variáveis Utilizadas . . . 427
58.2 Perfil de Caso . . . 427
58.3 Perfil de Grupo . . . 428
58.4 Distâncias Usadas . . . 428
58.5 Construindo uma Tipologia Inicial . . . 429
58.6 Características de Distâncias por Grupos . . . 430
58.7 Estatísticas de Resumo para Variáveis Quantitativas e para Variáveis Qualitativas Ativas . . . 431
58.8 Descrição de Tipologia Resultante . . . 431
58.9 Resumo da Quantidade de Variância Explicada pela Tipologia . . . 432
58.10Classificação Ascendente Hierárquica . . . 432
58.11Referências . . . 433

Apêndice: Mensagens de Erro dos Programas do IDAMS 435

Índice 437

Capítulo 1

Introdução

IDAMS é um pacote de programas para a validação, manipulação e análise estatística de dados. Ele está organizado como uma coleção de facilidades para gerenciamento e análise de dados acessíveis através de uma interface de usuário e de uma linguagem de controle comum. Exemplos dos tipos de dados que podem ser processados com IDAMS são: as respostas de entrevistados a questões em um survey, informação sobre livros em uma biblioteca, características pessoais e performace dos estudantes em um colégio, medidas advindas de um experimento científico. As características comuns a todos esses dados são que eles constituem valores de variáveis para cada coleção de objetos/casos (e.g. em uma pesquisa de survey, as questões correspondem às variáveis e os respondentes aos casos).

Existem muitos pacotes e programas diferentes para auxílio na análise estatística de tais dados. Uma característica especial do IDAMS é que ele proporciona facilidades para validação extensiva de dados (e.g. checagem de código e checagem de consistência) antes de embarcar na análise. No que diz respeito à análise, IDAMS executa técnicas clássicas como construção de tabelas, análise de regressão, análise de variância, análise de conglomerados e discriminante como também técnicas mais avançadas como análise de fatores de componentes principais e análise de correspondências, scoring baseado em ordenação parcial de casos, ordenação de alternativas, segmentação e tipologia iterativa. Além disso, WinIDAMS propicia meios para a construção interativa de tabelas multidimensionais, exploração interativa de dados e análise interativa de séries temporais.

1.1 WinIDAMS Interface do Usuário

É uma interface de múltiplos documentos (MDI) que permite trabalhar simultaneamente com diferentes tipos de documentos em janelas diferentes.

A Interface proporciona o seguinte:

- definição dos folders Dados, Trabalho e Temporário para uma aplicação;
- janela Dicionário para criação/atualização e exibição de arquivos Dicionário;
- janela Dados para criação/atualização e exibição de arquivos Dados;
- janela Setup para preparar/exibir arquivos Setup;
- janela Resultados para exibir, copiar e imprimir partes selecionadas dos resultados;
- editor de texto geral;
- uma opção para executar setups do IDAMS a partir de um arquivo ou de uma janela Setup ativa;
- facilidades interativas de importação/exportação de dados;
- acesso interativo aos componentes de análise de dados (Tabelas multidimensionais, GraphID, TimeSID);
- acesso on-line ao Manual de Referência.

1.2 Facilidades para Gerenciamento de Dados

Agregação de dados (AGGREG). Permite o agrupamento de registros de um número de casos em um único registro e produz um novo dataset com um registro para cada grupo. Por exemplo, registros representando membros de um domicílio são agrupados em um domicílio representando o registro. As variáveis nos novos registros são estatísticas descritivas dos registros individuais, e.g., soma, média, valor mínimo/máximo.

Construção de um dataset IDAMS (BUILD). Um arquivo de dados original (que pode conter registros múltiplos por caso) é carregado juntamente com um dicionário descrevendo as variáveis a serem selecionadas. BUILD checa a existência de valores não-numéricos em campos numéricos; campos em branco podem ser recodificados com valores numéricos especificados pelo usuário e outros valores não-numéricos são reportados e substituídos por 9's.

Verificação de códigos (CHECK). Reporta casos que tenham valores inválidos de variáveis. Códigos válidos para cada variável são especificados pelo usuário e/ou tirados do dicionário.

Verificação de consistência (CONCHECK). Reporta casos que possuam inconsistências entre duas ou mais variáveis. As declarações Recode do IDAMS são utilizadas para especificar as relações lógicas a serem checadas.

Verificação de intercalação de registros (MERCHECK). Confere se os registros corretos estão presentes para cada caso em um arquivo com múltiplos registros por caso. Ele produz um arquivo contendo um número igual de registros por caso. Registros inválidos ou duplicados podem ser deletados e registros perdidos podem ser inseridos com valores perdidos (missing values) especificados pelo usuário.

Correção de dados (CORRECT). Atualiza um arquivo Dados pela aplicação de correções aos valores de variáveis individuais em casos especificados. O arquivo Results contém a seqüência escrita das correções que possibilita que as mesmas sejam arquuivadas.

Importação/exportação de dados (IMPEX). Import objetiva a construção de dataset ou matrizes IDAMS de arquivos advindos de outros softwares. O objetivo de export é tornar possível o uso de arquivos Dados e Matrix, armazenados ou criados pelo IDAMS, por outros pacotes. Arquivos texto em formato livre ou DIF podem ser importados ou exportados.

Lista de datasets (LIST). Valores de variáveis selecionadas (originais ou recodificadas) e/ou casos selecionados podem ser listados em formato de coluna.

Intercalação de datasets (MERGE). Dois datasets podem ser fundidos utilizando-se de casos de similaridade de acordo com um conjunto de variáveis chamadas de variáveis de emparelhamento. Existem 4 opções para selecionar casos para o dataset resultante: (1) apenas casos presentes em ambos os arquivos (interseção); (2) casos presentes em qualquer arquivo (união); (3) todo caso presente no primeiro arquivo; (4) todo caso do segundo arquivo. O usuário especifica quais variáveis de cada um dos dois arquivos de entrada devem ser selecionadas. Existe uma opção para casar um caso de um arquivo com mais de um caso de um segundo arquivo, e.g. para adicionar dados domiciliares de um arquivo em cada registro individual em um segundo arquivo.

Classificação e fusão de arquivos (SORMER). Essa é uma ferramenta de finalidade geral para classificação de dados em ordem crescente ou decrescente em até 12 campos. Até 16 arquivos podem ser fundidos.

Subdivisão de datasets (SUBSET). Produz um novo dataset (arquivos Dados e Dicionário) contendo casos selecionados e/ou variáveis do dataset original. Há uma opção para checar a existência de dados duplicados.

Transformação de dados (TRANS). Permite que variáveis criadas com a ferramenta Recode do IDAMS possam ser salvas em um dataset permanente.

1.3 Facilidades para Análise de Dados

Análise de agrupamento (CLUSFIND). Executa análise de agrupamento via partição de um conjunto de objetos (casos ou variáveis) em um conjunto de clusters assim determinados por um dentre 6 algoritmos, 2 baseados na partição ao redor de medoids (objetos representativos), um baseado em agrupamento difuso e os outros 3 baseados em agrupamento hierarquizado.

Análise de configuração (CONFIG). Executa análise em uma configuração de entrada única, criada por exemplo pelo programa MDSCAL. Ele possui a capacidade de centrar, normalizar, rotar, transladar as dimensões, computando distâncias entre pontos e produtos escalares. A configuração pode ser impressa após cada transformação.

Análise discriminante (DISCRAN). Procura pela(s) melhor(es) função(ções) linear(es) de um conjunto de variáveis que reproduz, tanto quanto possível, um agrupamento a priori dos casos. É utilizado um procedimento passo-a-passo, i.e. em cada passo a variável mais poderosa é incluída. Três amostras de casos podem ser distinguidas: amostra básica, na qual os principais passos de análise discriminante são executados; amostra de teste, na qual a potência da função discriminante é checada; amostra anônima, a qual é utilizada apenas para classificar os casos. Escolha de casos e valores dos dois primeiros fatores discrimantes (se houver mais de 2 grupos) podem ser salvos em um dataset.

Funções de distribuição e de Lorenz (QUANTILE). Funções de distribuição apresentando entre 2 e 100 subintervalos, funções de Lorenz, curva de Lorenz e coeficientes de Gini, e o teste de Kolmogorov-Smirnov.

Análise fatorial (FACTOR). Cobre um conjunto de análises de componentes principais (produtos escalares, covariâncias, correlações) e correspondências de análise fatorial. Para cada análise, ele constrói uma matriz representando as relações entre variáveis e computa seus valores e vetores próprios. Após isto, ele calcula os fatores do caso e/ou da variável, dando para cada caso e/ou variável sua ordenada, qualidade de representação e contribuição para os fatores. Os fatores podem ser salvos em um dataset e uma representação gráfica dos casos e/ou variáveis no espaço de fatores pode ser obtida. Variáveis ativas e passivas podem ser distinguidas.

Regressão linear (REGRESSN). Análise de regressão linear múltipla: padrão e stepwise (passo a passo). Tanto um dataset quanto uma matriz de correlação podem ser usados como input. Resíduos podem ser impressos juntamente com a estatística de Durbin-Watson de autocorrelação de primeira ordem, e eles também podem ser salvos para futura análise.

Escalonamento multidimensional (MDSCAL). Esse é um procedimento de escalonamento multidimensional não-métrico para a análise de similaridades. Ele opera em uma matriz de medida de similaridade ou dissimilaridade e procura pela melhor representação geométrica dos dados no espaço n-dimensional. O usuário controla a dimensionalidade da configuração obtida, a métrica de distância e a maneira com que os empates (valores iguais) devem ser tratados.

Análise de classificação múltipla (MCA). Examina as relações entre diversos preditores (variáveis de controle) e uma única variável dependente, e determina o efeito de cada preditor antes e depois do ajustamento das inter-correlações com outros preditores. Propicia informação sobre relações bivariadas e multivariadas entre preditores e a variável depenedente. Resíduos podem ser impressos e/ou salvos em um dataset.

Análise de variância multivariada (MANOVA). Executa análise de variância e de covariância multivariada e univariada, usando um modelo linear geral. Até oito fatores (variáveis independentes) podem ser utilizados. Se mais de uma variável dependente for especificada, ambas as análises univariada e multivariada são executadas. O programa executa uma solução exata com números iguais ou desiguais de casos nas células.

Análise de variância univariada (ONEWAY). Estatísticas descritivas da variável dependente ao longo das categorias da variável de controle e estatísticas de análise, como: soma dos quadrados total, soma dos quadrados entre médias, soma dos quadrados entre grupos, eta e eta quadrado (ajustado e não-ajustado) e valor do F-test.

Scoring baseado em ordenação parcial de casos (POSCOR). Calcula os escores de escala ordinal de variáveis de intervalo ou de escala ordinal. Os escores são calculados para cada caso envolvido na análise e eles medem a posição relativa do caso no conjunto de casos. Os scores, opcionalmente com outras variáveis especificadas pelos usuários, são produzidos na forma de um dataset IDAMS.

Correlação de Pearson (PEARSON). Calcula os coeficientes de correlação r de Pearson, covariâncias e coeficientes de regressão. Pode ser requisitado que dados perdidos sejam deletados aos pares ou por casos. Matrizes de covariância e correlação produzidas podem ser salvas em um arquivo.

Ordenamento de alternativas (RANK). Determina um ordenamento razoável de alternativas usando dados de preferência e três procedimentos de ranking diferentes, um baseado em lógica clássica e os outros dois baseados em lógica difusa. Dados de preferência podem tanto representar uma seleção de alternativas ou ranking. Dois tipos de relações de preferências individuais podem ser especificadas: fraca e estrita. Com o ranking difuso, os dados determinam completamente os resultados obtidos, enquanto que com ranking

clássico o usuário tem a possibilidade de controlar os cálculos.

Diagramas de dispersão (SCAT). Diagramas de dispersão, estatísticas univariadas (média, desvio-padrão e N) e estatísticas bivariadas (r de Pearson e estatísticas de regressão: coeficiente B e constante A).

Busca de estrutura (SEARCH). Um procedimento de segmentação binária para desenvolver modelos preditivos. A questão "que partição dicotômica e em qual variável de previsão dará a maior melhoria na habilidade de prever os valores da variável dependente" embutida em um esquema iterativo, é a base do algoritmo usado.

Tabelas univariadas e bivariadas (TABLES). As opções incluem: (1) freqüência univariada acumulada e simples e distribuição percentual; (2) estatistícas univariadas: média, mediana, moda, variância, desvio-padrão, assimetria, curtose, mínimo e máximo; (3) tabelas de freqüência bivariada com percentuais totais, por linha e coluna; (4) tabelas com valores médios de uma variável adicional; (5) estatísticas bivariadas: teste-t para médias entre duas linhas, chi-quadrado, coeficiente de contigência, V de Cramer, Taus de Kendall, Gama, Lambda, ro de Spearman, um número de estatísticas para Medicina Baseada em Evidência, e três testes não-paramétricos: Wilcoxon, Mann-Whitney e Fisher.

Tipologia e classificação ascendente (TYPOL). Cria uma variável de tipologia como um resumo de um grande número de variáveis quantitativas e qualitativas. O usuário escolhe o número inicial e final de grupos, o tipo de distância utilizada, e a maneira que a tipologia inicial é iniciada. Os grupos da tipologia inicial são estabelecidos utilizando-se de um procedimento iterativo. O número de grupos pode ser reduzido usando-se um algoritmo de classificação ascendente hierárquica. Uma distinção pode ser feita entre variáveis ativas que participam na construção da tipologia, e variáveis passivas, para as quais as principais estatísticas são calculadas entre os grupos da tipologia.

Tabelas multidimensionais interativas. Esse componente permite visualizar e adequar tabelas multidimensionais com freqüências, percentuais de linha, de coluna e totais, estatísticas sumárias (soma, contagem, máximo, mínimo, variância, desvio-padrão) de variáveis adicionais, e estatísticas bivariadas. Até sete variáveis podem ser aninhadas em linhas e colunas. A construção de uma tabela pode ser repetida para cada valor até três variáveis de "página". As tabelas podem ser impressas também, ou exportadas em formato livre (delimitado por vírgula ou outro caracter) ou em formato HTML.

Exploração gráfica de dados interativa. Um componente separado, GraphID, está disponível para exploração de dados através de mostradores gráficos. O mostrador básico é em forma de múltiplos diagramas de dispersão para diferentes pares de variáveis. Informações adicionais como histogramas e linhas de regressão podem ser mostradas em cada diagrama. Os diagramas podem ser manipulados de várias maneiras. Por exemplo, casos selecionados podem ser marcados em um diagrama e então evidenciados em todos os outros diagramas. Partes dos mostradores podem ser aumentados ("zoomed"). As matrizes do IDAMS são mostradas como diagramas tridimensionais com as linhas e colunas sendo representadas por dois dos eixos e a terceira dimensão sendo usada para exibir o tamanho da estatística para cada célula.

Análise de séries temporais interativa. Outro componente separado, TimeID, propicia a análise interativa de séries temporais. Ele contém análise de tendência, auto-correlações e correlações cruzadas, análise estatística e gráfica de valores de séries temporais, testes de randomicidade e tendência, previsão no curto prazo, periogramas e estimação de densidades espectrais. Séries podem ser transformadas pelo cálculo de médias, composições aritméticas, diferenças seqüênciais, taxas de mudança, alisadas via médias móveis e decompostas usando filtros de freqüências.

1.4 Dados em IDAMS

Dataset IDAMS - o arquivo Dados. O arquivo de dados de entrada no IDAMS pode ser qualquer arquivo de formato fixo com caracteres (ASCII), i.e. os valores para uma dada variável ocupam a mesma posição (campo) no registro para cada caso. As características desse arquivo são:

- 1-50 registros por caso;
- cada caso pode conter até 4096 caracteres;
- número de casos limitado pela capacidade do disco e representasão interna de números;
- variáveis podem ser numéricas (até 9 caracteres) ou alfabético (até 255 caracteres).

Dataset IDAMS - o arquivo Dicionário. O dicionário é usado para descrever dados:

- pode conter até 1000 variáveis identificadas por um único número entre 1 e 9999;
- para cada variável, deve conter pelo menos o número da variável, o seu tipo (numérico ou alfanumérico), a sua localização no registro do dado;
- para cada variável, podem ser especificados, o nome da variável, dois códigos de valores perdidos, o número de casas decimais e um número de referência;
- para variáveis qualitativas, códigos e etiquetas correspondentes podem ser incluídas.

O par de arquivos consistindo de um arquivo Dicionário e um arquivo Dados descrito pelo primeiro é conhecido como um **dataset IDAMS.**

Matrizes em IDAMS. Alguns programas de análise usam uma matriz quadrada ou retangular como input, ao invés de dados não trabalhados.

A ***matriz quadrada*** é usada para arranjos simétricos de estatísticas bivariadas com uma constante na diagonal. Somente o canto superior direito da matriz é armazenado, sem a diagonal.

A ***matriz retangular*** é usada para arranjos de valores não simétricos. O significado das linhas e colunas varia de acordo com o programa do IDAMS.

1.5 Comandos do IDAMS e Arquivo "Setup"

Com exceção dos componentes interativos do WinIDAMS, a execução de progamas do IDAMS é iniciada por um setup. O setup contém informação como especificações de arquivo, declarações de controle de programas, instruções de codificação de variáveis, etc., separada por comandos do IDAMS (começados pelo símbolo $), que identifica o tipo de informação sendo especificada. O primeiro comando em um arquivo Setup do IDAMS sempre identifica o primeiro programa a ser executado, e.g.

```
$RUN TABLES
$FILES
DICTIN = nome do arquivo Dicionário
DATAIN = nome do arquivo Dados
$SETUP
    declaração de controle do programa TABLES
$RECODE
    declarações de codificação de variáveis
```

1.6 Características Padrão do IDAMS

Seleção de casos. Por default todos os casos de um arquivo Dados serão processados em uma execução de programa. Para selecionar um subconjunto, uma declaração de filtro é incluída no setup, e.g. INCLUDE V3=1 (inclui apenas aqueles casos onde a variável 3 é igual a 1).

Seleção de variáveis. Variáveis são referenciadas pelos números declarados no dicionário. Um grupo de variáveis é especificado em uma lista de variáveis depois de palavras-chave como VARS, CONVARS, OUTVARS. Essas listas de variáveis podem incluir também variáveis-R construídas pela facilidade Recode do IDAMS (ver abaixo), e.g. VARS=(V3-V6,V129,R100,R101).

Transformando/recodificando dados. Uma facilidade poderosa do Recode permite a recodificação de variáveis e a construção de novas variáveis. Instruções de recodificação são preparadas pelo usuário na linguagem Recode do IDAMS. Isso inclui a possibilidade de computação aritmética como também o uso de várias funções especiais para operações como agrupamento de valores, a criação de variáveis "dummy", etc. Declarações condicionais são também permitidas. Exemplos de declarações de Recode para a construção de 3 novas variáveis são:

```
R100=V4+V5
R101=BRAC(V10,0-15=1,16-60=2,61-98=3,99=9)
IF (MDATA(V3,V4) OR V4 EQ 0) THEN V102=99 ELSE R102=V3*100/V4
```

As variáveis-R construídas para cada caso podem ser usadas temporariamente no programa em execução ou podem ser salvas em um dataset usando o programa TRANS.

Ponderando dados. Quando uma estrutura de amostragem complexa é utilizada durante a coleta, pode ser necessário usar pesos diferentes para casos durante a análise. Tais pesos são usualmente armazenados como uma variável em um arquiuvo Dados. O parâmetro WEIGHT é então usado nas declarações de controle do programa para invocar a ponderação, e.g. WEIGHT=V5.

Tratamento de dados perdidos e dados "ruins". Valores especiais para cada variável numérica podem ser identificados como dados perdidos e armazenados em um dicionário. Durante o processamento de dados, os dados perdidos são manuseados através de dois parâmetros:

- MDVALUES (especifíca quais códigos de dados perdidos devem ser usados para checar a ocorrência de dados perdidos em variáveis numéricas);
- MDHANDLING(específica o que deve ser feito se forem encontrados dados perdidos).

Normalmente se assume que os dados foram limpos antes da análise. Se esse não for o caso, então o parâmetro BADDATA está disponível para pular casos com valores não-numéricos (incluindo campos em branco) em campos numéricos, ou para tratar tais valores como dados perdidos.

1.7 Importação e Exportação de Dados

IDAMS não usa nenhum formato de arquivo interno especial para armazenar dados. Qualquer arquivo de caracteres em formato fixo pode ser descrito por um dicionário IDAMS e então ser carregado no IDAMS. Diferentemente, dados em formato livre com Tab, vírgula ou ponto-e-vírgula usados como separadores podem ser importados através da Interface de Usuário do WinIDAMS. Além disso, o programa IMPEX permite que um arquivo em formato fixo do IDAMS seja criado a partir de qualquer arquivo de texto em formato livre ou formato DIF.

Arquivos de dados criados pelo IDAMS são sempre arquivos de caracteres em formato fixo. Tais arquivos podem ser usados diretamente por outros softwares em conjunto com a informação descritiva apropriada dos dados para aquele programa. Dados em formato livre com Tab, vírgula ou ponto-e-vírgula usados como separadores podem ser obtidos através da Interface de Usuário do WinIDAMS. Além disso, o programa IMPEX permite que um arquivo em formato fixo do IDAMS seja exportado como texto em formato livre ou formato DIF.

Matrizes IDAMS são armazenadas em um formato específico do IDAMS (descrito no capítulo "Dados em IDAMS"). O programa IMPEX pode ser usado para importar/exportar matrizes em formato livre.

1.8 Troca de Dados entre CDS/ISIS e IDAMS

Há um programa separado, WinIDIS, que prepara a descrição de dados e executa trasferência de dados entre IDAMS e CDS/ISIS (o software da UNESCO para gerenciamento de base de dados e recuperação de informação). Essa transferência é controlada pelos arquivos de descrição de dados do IDAMS e do ISIS (o dicionário do IDAMS e a Tabela de Definição de Campo do CDS/ISIS). Ao se ir do ISIS para o IDAMS, um novo dicionário e arquivos Dados são sempre construídos e podem ser fundidos com outros dados usando-se as facilidades de gerenciamento de dados do IDAMS. Ao se ir do IDAMS para o ISIS, existem três possibilidades: (1) uma base de dados completamente nova pode ser construída, (2) dados transferidos podem ser adicionados a uma base de dados já existente como registros novos, (3) registros de uma base de dados já existente podem ser atualizados com os dados transferidos.

1.9 Estrutura deste Manual

Todas as características gerais do IDAMS, incluindo a facilidade Recode, são descritas na ***Parte 1*** deste Manual.

Parte 2 inclui instruções de instalação, descrição dos arquivos e folders usados no WinIDAMS, uma seção intitulada "Iniciando" que conduz o usuário pelos passos necessários para executar tarefas simples, e descrição da Interface do Usuário do WinIDAMS.

Descrições aprofundadas de cada programa do IDAMS são dadas nas ***Partes 3 e 4***. Essas descrições contêm as seguintes seções:

Descrição Geral. Uma descrição dos propósitos básicos do programa.

Características Padrão do IDAMS. Declarações sobre as possibilidades de seleção de caso e variável, transformação de dados, capacidades de ponderação e manuseio de dados perdidos.

Resultados. Detalhes dos resultados a serem impressos (ou revisados na tela).

Descrição dos arquivos de entrada e saída. Uma seção para cada dataset IDAMS, cada matriz e cada arquivo de entrada ou saída, dando a descrição dos seus conteúdos.

Estrutura de Setup. Uma designação das especificações do arquivo, comandos do IDAMS e declarações de controle do programa necessárias para executar o programa.

Declarações de Controle de Programa. Os parâmetros e/ou formatos de cada declaração de Controle do Programa, com um exemplo para cada tipo.

Restrições. Um resumo das limitações do programa.

Exemplos. Exemplos de conjuntos completos de declarações de controle para executar o programa.

Parte 5 dá a descrição dos componentes interativos do WinIDAMS para a construção de tabelas multidimensionais, para a exploração gráfica de dados e análise de séries temporais.

Parte 6 propicia os detalhes das técnicas estatísticas, fórmulas e referências bibliográficas para todos os programas de análise.

Finalmente erros sinalizados pelos programas do IDAMS aparecem resumidos no Apêndice.

Parte I

Noções Fundamentais

Capítulo 2

Dados em IDAMS

2.1 O Dataset IDAMS

2.1.1 Descrição Geral

O dataset consiste de dois arquivos separados: um arquivo Dados e um arquivo Dictionário que descreve alguns ou todos os campos (variáveis) nos registros do arquivo de dados. Todos os arquivos Dictionário/Dados produzidos pelos programas do IDAMS são datasets IDAMS.

2.1.2 Método de Armazenamento e Acesso

Ambos os arquivos Dictionário e Dados são lidos e escritos sequencialmente. Portanto, eles podem ser salvos em qualquer mídia. Não existe um arquivo "sistema" especial interno de IDAMS como em alguns pacotes. Os arquivos são em caracteres/texto (ASCII) e podem ser processados a qualquer tempo com editores gerais, ou alimentados diretamente em outros pacotes estatísticos.

2.2 Arquivos Dados

2.2.1 O arranjo de Dados

A despeito do formato real do arquivo de dados, os dados podem ser visualizados como um arranjo retangular dos valores das variáveis, onde o elemento x_{ij} é o valor da variável representada pela j-ésima coluna do caso representado pela i-ésima linha. Por exemplo, os dados de um survey podem ser mostrados da seguinte maneira:

Casos	Variáveis				
	identificação	educação	sexo	idade	...
caso 1	1300	6	2	31	... ...
caso 2	1301	2	1	25	...
.	1302	3	1	55	...
.	.	.	.	.	...

Nesse exemplo, cada linha representa um respondente em um survey e cada coluna representa um item do questionário.

2.2.2 Características do Arquivo Dados

Esses arquivos contém normalmente, mas não necessariamente registros de comprimento fixo, dado que o final do registro é reconhecido através dos caracteres retorno/nova linha. Contudo, o comprimento do registro mais longo deve ser informado na definição do arquivo (ver o comando $FILES). Não há limite no número de registros no arquivo Dados.

O comprimento máximo de cada registro é de 4096 caracteres.

Cada "caso" pode consistir de mais de um registro (até um máximo de 50). Se, na execução de um programa em particular, variáveis devem ser acessadas a partir de mais de um tipo de registro, então deve haver o mesmo número de registros para cada caso. O programa MERCHECK pode ser usado para criar arquivos que estejam de acordo com essa condição. Note que qualquer arquivo Dados produzido por um programa do IDAMS é sempre reestruturado para conter um único registro por caso.

Se um arquivo de dados brutos contém diferentes tipos de registros (e o tipo de registro está codificado) e não possui exatamente o mesmo número de registros por caso, os programas do IDAMS podem ser executados usando-se variáveis de um tipo de registro por vez através da seleção, no início, desse tipo de registro apenas.

2.2.3 Arquivos Hierárquicos

IDAMS processa apenas arquivos "retangulares" como descrito acima. Arquivos hierárquicos podem ser manipulados armazenando-se registros de diferentes níveis em arquivos diferentes e então usando-se os programas AGGREG e MERGE para produzir registros compostos contendo variáveis de níveis diferentes. Alternativamente, o arquivo hierárquico completo pode ser processado um nível por vez via "filtragem" dos registros para aquele nível apenas (considerando que os tipos dos registros foram codificados).

2.2.4 Variáveis

Referenciando variáveis. As variáveis no arquivo Dados são identificadas por um único número entre 1 e 9999. Esse número, precedido por um V (e.g. V3) é usado para se referir a uma variável particular em declarações de controle para programas. O número da variável é usado para indexar um registro definidor de variável no dicionário que fornece todas as outras informações necessárias sobre a variável, como seu nome e sua localização no registro de dados.

Tipos de variáveis. Variáveis podem ser do tipo numérica ou alfabética, ambas armazenadas em modo caracter.

Variáveis numéricas. Podem ter valores positivos ou negativos e possuem as seguintes características:

- Um valor pode ser composto de caracteres numéricos 0-9, um ponto decimal e um sinal (+,-). Espaços em branco no começo são permitidos.
- Valores devem ser alinhados a direita (i.e. sem espaços em branco no final) ao não ser que um ponto decimal apareça explicitamente.
- A largura máxima do campo é de 9 mas um número de dígitos significativos de até 7 apenas (contando os inteiros e decimais juntos) é mantindo no processamento.
- Valores de variáveis podem inteiros (e.g. a idade ou uma variável categórica como sexo) ou pode ser decimal (e.g. uma variável medindo percentuais). O número de casas decimais (NDEC) é armazenado no registro definidor de variável no dicionário. Normalmente o ponto decimal está "implícito" e não aparece nos dados. Nesse caso, NDEC dá o número de dígitos do valor da variável que devem ser tratados como casas decimais. Se um ponto decimal "explícito" é codificado nos dados, então NDEC é usado para determinar o número de dígitos a direita do ponto decimal que será retido, arredondando o valor caso necessário, e.g. valores 4.54 e 4.55 com NDEC=1 serão utilizados como 4.5 e 4.6, respectivamente.
- Um sinal (se ele aparecer) deve ser o primeiro caracter, e.g. "-0123".
- Campos em branco são considerados não-numéricos e tratados como dado "ruin". Veja abaixo como lidar com espaços em branco nos dados para indicar dados perdidos ou dados inapropriados.

- Com exceção de BUILD, todos os programas do IDAMS aceitam valores em notação exponencial, e.g. o valor codificado .215E02 será usado como 21.5.

Variáveis alfabéticas. Variáveis alfabéticas podem ser mantidas em arquivos Dados e podem ter um tamanho de até 255 caracteres. Elas podem ser usadas em programas de gerenciamento de dados. Variáveis alfabéticas de 1-4 caracteres podem ser usadas também como filtros. Para serem utilizadas na análise, essas variáveis devem ser salvas como numéricas. Isso deve ser feito com a função BRAC do Recode.

2.2.5 Códigos para Dados Perdidos

O valor da variável para um caso particular pode ser desconhecida por diversas razões, por exemplo uma questão pode ser não-aplicável para certos respondentes ou um respondente se nega a responder uma questão. Códigos de valores perdidos especiais podem ser estabelecidos para cada variável numérica e codificada nos dados quando necessário. Dois códigos de dados perdidos são permitidos: MD1 e MD2. Se usados, qualquer valor nos dados igual a MD1 é considerado um valor perdido; qualquer valor maior que ou igual a MD2 (se MD2 é positivo ou zero) ou menor que ou igual (se MD2 é negativo) é também considerado perdido.

Esses códigos de dados perdidos são armazenados no registro de dicionário para a variável. Similarmente a valores dos dados, eles podem ser inteiros ou decimais, com um ponto decimal implícito ou explícito. Se MD1 ou MD2 é especificado com um ponto decimal implícito, NDEC dá o número de dígitos a ser tratado como casas decimais. Se um ponto decimal explícito é codificado em MD1 ou MD2, então NDEC determina o número de dígitos a direita do ponto decimal que deve ser mantido, arredondando o valor de acordo com o exposto.

Quando os códigos MD1 e MD2 de uma variável estão em branco no dicionário, isso significa que não há códigos de dados perdidos especiais. Durante a execução de um programa do IDAMS, os campos MD1 e MD2 em branco no dicionário são preenchidos com os valores default de dados perdidos de 1.5×10^9 and 1.6×10^9 respectivamente.

Como os códigos de dados perdidos são limitados a um máximo de 7 dígitos (ou 6 dígitos e um sinal negativo), eles podem apresentar um problema para variáveis de 8 e 9 dígitos. O usuário deve considerar o uso de um sinal negativo no código de dado perdido nesse caso.

2.2.6 Valores Não-numéricos ou em Branco em Variáveis Numéricas - Dados Ruins

Nos programas para gerenciamento de dados do IDAMS, valores de dados são meramente copiados de um lugar para outro, e a conversão para um modo computacional (binário) não é realizado; nesse caso não há uma checagem se variáveis numéricas possuem valores numéricos. Contudo, quando variáveis estão sendo utilizadas em análises ou em operações de Recode, nesse caso seus valores são convertidos para modo binário e valores contendo caracteres não-numéricos causarão problemas. Normalmente tais caracteres devem ser limpos dos dados antes da análise. Além disso, valores em branco em variáveis numéricas não são tratados automaticamente como dados perdidos; eles também são considerados como dados não-numéricos ou dados "ruins".

Para permitir a análise de dados com limpeza incompleta e para o manuseio de campos em branco não codificados, o parâmetro BADDATA pode ser usado para tratar valores em branco e outros valores não-numéricos como valores perdidos e, portanto, ter a possibilidade de eliminá-los da análise. Especificação do parâmetro BADDATA=MD1 ou BADDATA=M2 resulta na conversão de valores "ruins" para o código MD1 ou MD2 da variável. Se os códigos do MD1 ou MD2 estão em brancos, os códigos de dado perdido são convertidos para os valores default correspondentes (ver acima) e, portanto, são tratados como dados perdidos (ver a descrição do parâmetro BADDATA no capítulo "O Arquivo Setup do IDAMS").

2.2.7 Regras de Edição de Variáveis Produzidas pelos Programas do IDAMS

Os programas IDAMS sempre criam um arquivo Dados e um correspondente dicionário IDAMS, i.e. um dataset IDAMS.

O arquivo Dados contém um registro para cada caso. O comprimento do registro é a soma da largura do campo de todas as variáveis produzidas e é determinada pelo programa.

Valores de variáveis numéricas são editados em uma forma padrão descrita abaixo:

- Se o campo inteiro contém apenas os caracteres numéricos 0-9, eles serão processados exatamente como eles aparecem nos dados de entrada.
- Se o campo contém um número com espaços no começo (e.g. ' 5'), os espaços em branco são convertidos em zeros antes dos dados serem processados. Campos com espaços no final (e.g. '04 ' em um campo númerico de três dígitos), com espaços em branco intermediários (e.g. '0 4') e totalmente em branco são tratados de acordo com a especificação de BADDATA.
- Se o campo contém um valor positivo ou negativo com os caracteres '+' e '-' explicitamente incluídos, o valor positivo é removido e o sinal negativo é colocado antes do primeiro dígito numérico significante.
- Se o campo contém um número com um ponto decimal explícito, este é removido e o valor produzido tem a mesma largura do campo original e n casas decimais como definido no campo NDEC de descrição de variável. Espaços em brancos no início do campo são convertidos em zeros. Se mais de n dígitos são encontrados no campo de entrada depois do ponto decimal, o valor é arredondado e convertido em n casas decimais (e.g. if n=2, um valor de entrada de 2.146 será convertido em 2.15; se n=0, um valor de entrada de 1.5 será convertido em 002). Espaços em branco no final não causam condição de erro. Se menos que n dígitos são encontrados, zeros são inseridos a direita das casas decimais que faltam.
- Valores grandes demais para caber no campo especificado são tratados de acordo com a especificação contida em BADDATA.

Valores de variáveis alfabéticas não são editados e são os mesmos na entrada e saída.

2.3 O Dicionário do IDAMS

2.3.1 Descrição Geral

O dicionário é usado para descrever as variáveis nos dados. Para cada variável ele deve conter no mínimo o número da variável, seu tipo e sua localização no registro dos dados. Além disso, um nome de variável, dois códigos de valores perdidos, o número de casas decimais e um número de referência ou nome devem ser dados. Essa informação é armazenada em registros de descrição de variáveis às vezes conhecidos como registros-T. Registros-C opcionais para variáveis categóricas definem labels para códigos possíveis diferentes. O primeiro registro no dicionário, o registro de descrição de dicionário, identifica o tipo de dicionário, dá o primeiros e últimos números da variável usados no dicionário e especifica o número de registros de dados que compõe um "caso".

O dicionário original é preparado pelo usuário para descrever os dados brutos. Os programas do IDAMS que produzem datasets sempre produzem novos dicionários refletindo o novo formato dos dados criados.

Registros do dicionário possuem formato fixo e comprimento de 80-caracteres.

Um descrição detalhada de cada tipo de registro de dicionário é dada abaixo.

Registro de descrição de dicionário. Esse é sempre o primeiro registro no dicionário.

Colunas	*Conteúdo*
4	3 (indica o tipo de dicionário).
5-8	Número da primeira variável (alinhamento a direita).
9-12	Número da última variável (alinhamento a direita).
13-16	Número de registros por caso (alinhamento a direita).
20	Forma em que a localização da variável é especificada (colunas 32-39) nos registros de descrição de variáveis.
	branco Número do registro e colunas inicial e final. O comprimento do registro deve ser 80 para usar esse formato se o número de registros por caso é > 1.
	1 Localização inicial e largura do campo.

Registros de descrição de variáveis (Registros-T). O dicionário contém um desses registros para cada variável. Esses registros são organizados em ordem crescente do número da variável. Os números das variáveis não precisam ser contíguos. O número máximo de variáveis é 1000.

Colunas	*Conteúdo*
1	T
2-5	Número da variável.
7-30	Nome da variável.
32-39	Localização; de acordo com a coluna 20 do registro de descrição de dicionário.
	Este
	32-33 Número da seqüência do registro contendo a coluna inicial da variável.
	34-35 Número da coluna inicial.
	36-37 Número da seqüência do registro contendo a coluna final da variável.
	38-39 Número da coluna final.
	Ou este
	32-35 Localização inicial da variável no caso.
	36-39 Largura do campo (1-9 para variáveis numéricas e 1-255 para variáveis alfabéticas).
40	Número de casas decimais (apenas valores numéricos). Espaços em branco não implicam em casas decimais.
41	Tipo de variável.
	branco Numérica.
	1 Alfabética.
45-51	Primeiro código de dados perdidos para variáveis numéricas (ou espaços em branco caso não se especifique o primeiro código de dados perdidos). Alinhamento a direita.
52-58	Segundo código de dados perdidos para variáveis numéricas (ou espaços em branco caso não se especifique o segundo código de dados perdidos). Alinhamento a direita.
59-62	Número de referência (opcional - pode ser usado para conter alguma referência alfanumérica imutável para a variável, e.g. o número original da variável ou referência a uma questão).
73-75	ID do estudo (opcional - pode ser usado para identificar o estudo ao qual esse dicionário pertence).

Nota 1: Quando números de registro e colunas são usados para indicar a localização da variável, listagens dos registros do dicionário não mostram os números de registro e colunas do jeito que eles aparecem no registro do dicionário. Ao contrário, a localização da variável é traduzida e impressa em um formato localização inicial/largura. Por exemplo, para uma variável nas colunas 22-24 do terceiro registro de um arquivo de dados de múltiplos registros (comprimento do registro de 80) por caso, a localização inicial será 182 (2 * 80 + 22) e a largura 3.

Nota 2: Se há mais de um registro por caso e o comprimento do registro não é 80, então a notação da localização inicial e da largura do campo devem ser usadas nos registros-T. A localização inicial é contada a partir do início do primeiro registro. Por exemplo, para registros de comprimento 121, a localização inicial do campo na posição 11 do segundo registro para o caso seria 132.

Registros de código-label (Registros-C). O dicionário pode opcionalmente conter esses registros para quaisquer variáveis. Eles seguem imediatamente depois do registro-T da variável a que eles se aplicam e propiciam códigos e suas labels para diferentes valores possíveis da variável. Eles são utilizados em programas como TABLES para imprimir as labels da linha e da coluna juntamente dos códigos correspondentes. Eles também podem ser usados como a especificação de códigos válidos para uma variável durante a entrada de

dados na Interface do Usuário do WinIDAMS com o programa CHECK.

Colunas	*Conteúdo*
1	C
2-5	Número da variável.
6-9	Número de referência (opcional - pode ser usado para conter alguma referência alfanumérica imutável, e.g. o número de variável original ou a referência a uma questão).
15-19	Valor do código alinhado a **esquerda**.
22-72	Label para esse código. (Note que apenas os 8 primeiros caracteres serão utilizados por programas de análise imprimindo labels de código, apesar de que a label completa aparecerá nas listagens do dicionário).
73-75	ID do estudo (opcional).

2.3.2 Exemplo de um Dicionário

```
Colunas:          1         2         3         4         5         6...
         12345678901234567890123456789012345678901234567890123456789012345678901...

            3   1  20   1   1
         T   1 Identificação              1   5
         T   2 Idade                      6   2          99
         T   3 Sexo                       8   1
         C   3         1      Feminino
         C   3         2      Masculino
         T  11 Região                    16   1
         C  11         1      Norte
         C  11         2      Sul
         C  11         3      Leste
         C  11         4      Oeste
         T  12 Média da nota             17   31        000    900
         T  20 Nome                      31  30 1
```

Isso é um dicionário descrevendo 6 campos de dados em um registro de dados mostrado diagramaticamente abaixo.

1-5	6-7	8		16	17-19		31-60	
V1 ID	V2 Idade	V3 Sexo		V11 Região	V12 Nota		V20 Nome	

As localizações das variáveis são expressas em termos de posição inicial e largura de campo (1 na coluna 20 do definidor do dicionário) e há um registro por caso (1 na coluna 16). Existe uma casa decimal na variável média da nota (V12). A variável idade possui um código 99 para dados perdidos. Para a média da nota, 0's implican dados perdidos, como também dados com valores maiores ou iguais 90.0. O nome de cada respondente (V20) é armazenado como uma variável de caracteres alfabéticos (tipo 1). Note que os números das variáveis não precisam ser contíguos e que nem todos os campos na base de dados precisa ser descrito.

2.4 Matrizes IDAMS

Existem dois tipos de matrizes IDAMS: quadrada e retangular. Ambos os tipos são auto-definidos, mas diferentemente dos datasets IDAMS, o "dicionário" é armazenado no mesmo arquivo do arranjo de valores. Em geral, essas matrizes são criadas por um programa do IDAMS para serem usadas como entrada em outro programa e, portanto, o usuário não precisa estar familiarizado com o formato. Se, contudo, for necessário preparar uma matriz de similaridade, uma matriz de configuração, etc. na mão, então os formatos descritos abaixo devem ser observados.

A despeito do tipo, todos os registros são fixos e com comprimento de 80 caracteres.

2.4.1 A Matriz Quadrada do IDAMS

A matriz quadrada pode ser utilizada apenas para um arranjo quadrado e simétrico. Apenas valores na parte superior direita triangular, fora da diagonal do arranjo é armazenada de fato na matriz quadrada. Uma matriz de coeficientes de correlação Pearsonianos é adequadamente armazenada dessa maneira.

Programas que usam/produzem matrizes quadradas. PEARSON produz matrizes quadradas de correlações e covariâncias; REGRESSN produz matrizes quadradas de correlações; TABLES produz matrizes quadradas de medidas de associação bivariada. Essas matrizes são inputs apropriados para outros programas, e.g. a matriz de correlação produzida por PEARSON pode ser utilizada em REGRESSN e em CLUSFIND. Além disso, CLUSFIND e MDSCAL utilizam matrizes quadradas de similaridades e dissimilaridades.

Exemplo.

```
          Colunas:                  111111111122222222223...
                                12345678901234567890123456789 0...

Definidor de matriz                  2   4
Declarações de formato |  #F  (12F6.3)
                       |  #F  (6E12.5)
Identificações de      |  #T   1 IDADE
              variável |  #T   3 EDUCAÇÃO
                       |  #T   9 RELIGIÃO
                       |  #T  10 SEXO
Arranjo de valores     |   -.011 -.174 -.033
                       |    .131 -.105
                       |   -.133
Médias & desvios-      |   0.33350E 01 0.54950E 01 0.50251E 01 0.40960E 01
              padrões  |   0.20010E 01 0.19856E 01 0.15000E 01 0.12345E 01
```

Formato. A matriz quadrada contém o seguinte:

1. Um registro definidor da matriz. Ele, o primeiro registro, dá o tipo de matriz e as dimensões do arranjo de valores.

Colunas	*Conteúdo*
4	2 (indica matriz **quadrada**).
5-8	O número de variáveis (alinhado a direita).

2. Uma declaração em formato Fortran descrevendo cada linha do arranjo de valores. A declaração de formato descreve o número de campos de valores em cada registro de 80 caracteres e o formato de cada um. Por exemplo, um formato (12F6.3) indica que cada linha do arranjo Dictionário é armazenada com até 12 valores por registro, cada valor ocupando 6 colunas, 3 das quais são casas decimais. Se uma linha contém mais de 12 valores, um novo registro conterá o 13-o valor, etc. Cada nova linha do arranjo começa sempre em um novo registro.

Colunas	*Conteúdo*
1-2	#F
3-80	A declaração de formato, entre parênteses.

3. Uma declaração em formato Fortran descrevendo os vetores das médias e desvios-padrões das variáveis. A declaração de formato descreve o número de valores por registro e o formato de cada um.

Colunas	*Conteúdo*
1-2	#F
3-80	A declaração de formato, entre parênteses.

4. Registros de identificação de variáveis. São n registros, onde n é o número de variáveis especificado no registro de definição da matriz. A ordem desses registros corresponde à ordem das variáveis que indexam as linhas (e colunas) do arranjo de valores. Quando uma matriz é criada por um programa do IDAMS, o número de variáveis e seus nomes são mantidos no dataset IDAMS de onde as estatísticas bivariadas são geradas.

Colunas	*Conteúdo*
1-2	#T ou #R (indica identificação de variáveis para uma linha da matriz).
3-6	O número da variável (alinhado a direita).
8-31	O nome da variável.

As quatro seções da matriz mostradas acima são referidas como o "dicionário" da matriz. O dicionário da matriz é seguido pelo arranjo de valores.

5. O arranjo de valores. Dado que o arranjo é simétrico e possui células na diagonal contendo constantes (e.g. uma correlação de 1.0 para uma variável correlacionada com ela mesma), apenas o canto superior direito, fora da diagonal, do arranjo é armazenado. Note que para a covariância da matriz os elementos da diagonal podem ser calculados usando-se desvios-padrões que estão incluídos no arquivo da matriz (ver ponto 7 abaixo).

   No exemplo da matriz de 4 variáveis acima, o arranjo completo (antes de entrar em formato quadrado) seria o seguinte

```
vars        1       3       9      10
 1      1.000   -.011   -.174   -.033
 3      -.011   1.000    .131   -.105
 9      -.174    .131   1.000   -.133
10      -.033   -.105   -.133   1.000
```

   A parte do arranjo que é armazenada é:

```
vars        1       3       9      10
 1              -.011   -.174   -.033
 3                       .131   -.105
 9                              -.133
10
```

   Cada linha desso arranjo reduzida inicia um novo registro e é escrito de acordo com a especificação de formato do dicionário da matriz (ver acima).

6. Um vetor de médias das variáveis. Os n valores são gravados de acordo com a declaração de formato do dicionário da matriz.

7. Um vetor de desvios-padrões das variáveis. Os n valores são gravados de acordo com a declaração de formato do dicionário da matriz.

2.4.2 A Matriz Retangular do IDAMS

A matriz retangular difere da matriz quadrada no sentindo de que o arranjo de valores pode ser quadrado (e não-simétrica) ou retangular. Além disso, dado que as linhas de algumos larranjos não são indexadas por variáveis, e.g. uma tabela de freqüência, a matriz retangular pode ou não conter registros de identificação de variáveis; a matriz retangular não possui médias de variáveis nem desvios-padrões de variáveis.

Programas que usam/produzem matrizes retangulars. Essas matrizes são criadas pelos programas CONFIG, MDSCAL, TABLES e TYPOL. Elas são uma entrada apropriada para CONFIG, MDSCAL e TYPOL.

Exemplo.

```
          Colunas:                    1111111111222222222 3...
                                 12345678901234567890123456789 0...

Definidor de matriz                 3   4   3
Declaração de formato             #F (l6F5.0)
Identificações de variável  | #T   2 QI
                            | #T   5 EDUCAÇÃO
                            | #T   8 MOBILIDADE
                            | #T  12 RIVALIDADE FRATERNA
Arranjo de valores          |    59   20   10
                            |    37   15    2
                            |    50   40    7
                            |     8   26   31
```

Formato. A matriz retangular contém o seguinte:

1. Um registro definidor de matriz.

Colunas	*Conteúdo*
4	3 (indica uma matriz **retangular**).
5-8	O número de linhas (alinhado a direita).
9-12	O número de colunas (alinahdo a direita).
16	Número de registros de declaração de formato (#F). (Espaços em branco implicam 1).
20	Presença de labels de linha e de coluna.
	branco/0 Labels de linha, apenas, estão presentes (registros #R ou #T).
	1 Labels de coluna, apenas, estão presentes (registros #C).
	2 Labels de linha e de coluna estão presentes (registros #R ou #T, e #C).
	3 Nem labels de linha nem de coluna estão presentes.
21-40	Nome da variável da linha (opcional).
41-60	Nome da variável da coluna (opcional).
61-80	Descrição do conteúdo da matriz (opcional):
	Weighted frequencies (freqüências ponderadas)
	Unweighted frqs (freqüências não-ponderadas)
	Row Percentages (percentuais das linhas)
	Column percentages (percentuais das colunas)
	Total percentages (percentuais totais)
	Nome da variável cujos valores médios estão incluídos na matriz.

2. Uma declaração em formato Fortran descrevendo cada linha do arranjo de valores. O formato descreve um registro de 80 caracteres. Por exemplo, um formato (16F5.0) indica que cada linha do arranjo é armazenada com até 16 valores por registro e com cada valor ocupando 5 colunas, nenhuma das quais é casa decimal.

Colunas	*Conteúdo*
1-2	#F
3-80	A declaração de formato, entre parênteses.

3. Registros de identificação de variáveis. A ordem desses registros corresponde à ordem das variáveis/códigos indexando as linhas e colunas da matriz. Quando uma matriz retangular é criada por um programa do IDAMS, os número e nomes da variável/código são retidos no dataset de entrada ou na matriz da qual o arranjo de valores foi derivado.

Colunas	*Conteúdo*
1-2	#T ou #R para labels de linha, #C para labels de coluna.
3-6	O número da variável ou valor do código (alinhado a direita).
	Os valores de código maiores que 4 caracteres são substituídos por ****.
8-58	O nome da variável ou a label do código.

   As três seções acima são referidas como o "dicionário" da matriz. Seguindo o dicionário da matriz vem o arranjo de valores.

4. O arranjo de valores. O arranjo completo é armazenado. Cada linha começa um novo registro e é escrita de acordo com o formato especificado no dicionário da matriz.

2.5 Uso de Dados de outros Programas

2.5.1 Dados Brutos

Qualquer dado com registros em formato fixo (ASCII) pode ser utilizado diretamente pelos programas do IDAMS. Praticamente todos os pacotes estatísticos e de base de dados possuem uma função de "exportação" o "conversão" para produzir arquivos de dados em modo de caracteres de formato fixo. Um dicionário do IDAMS deve ser preparado para descrever os campos requeridos pelos dados.

Arquivos de dados em formato livre com Tab, vírgula ou ponto-e-vírgula usados como separadores podem ser importados diretamente através da Interface do Usuário do IDAMS. Ver o capítulo "Interface do Usuário" para detalhes.

Arquivos de texto em formato livre (qualquer caracter sendo usado como delimitador, incluindo espaços em branco) e em formato DIF podem também ser importados usando o programa IMPEX.

Dados armazenados em uma base de dados CDS/ISIS podem ser importados pelo IDAMS usando-se o programa WinIDIS.

2.5.2 Matrizes

O programa IMPEX pode ser usado para importar matrizes em formato livre. Além disso, matrizes produzidas fora do IDAMS, por exemplo, uma matriz de uma publicação, pode também ser armazenada de acordo com o formato dado acima.

Capítulo 3

O Arquivo Setup do IDAMS

3.1 Conteúdo e Finalidade

Para executar programas IDAMS, o usuário prepara um arquivo especial chamado arquivo "Setup" que controla a execução dos programas. Esse arquivo contém declarações de controle e comandos do IDAMS necessários para execução do programa, como: referência ao programa a ser executado, os nomes dos arquivos, as opções a serem selecionadas para o programa e instruções de transformação das variáveis, e.g.

```
$RUN nome do programa
$FILES
    especificações de arquivo
$SETUP
    declarações de controle do programa
$RECODE
    declarações de Recode
```

3.2 Comandos do IDAMS

Esses comandos, que se iniciam com "$", separam os diferentes tipos de informação propiciados pela execução de um programa IDAMS. Comandos disponíveis são:

$RUN programa	(nome do programa a ser executado)
$FILES [RESET]	(sinaliza o início das especificações de arquivo)
$RECODE	(sinaliza o início das declarações de Recode)
$SETUP	(sinaliza o início das declarações de controle de programa)
$DICT	(sinaliza o início de dicionário)
$DATA	(sinaliza o início dos dados)
$MATRIX	(sinaliza o início de uma matriz)
$PRINT	(ativa ou desativa a impressão)
$COMMENT [texto]	(comentários)
$CHECK [n]	(checa se o passo anterior terminou bem).

A primeira linha em um arquivo Setup deve ser sempre um comando $RUN identificando o programa a ser executado. Outros comandos relacionados à execução desse programa (seguidos de declarações associadas de controle ou dados) podem ser colocados em qualquer ordem. Esses são então seguidos pelo comando $RUN para o próximo programa (se houver) para serem executados e assim sucessivamente. Os comandos individuais do IDAMS são descritos abaixo em ordem alfabética.

$CHECK [n]. Se esse comando está presente, o programa não será executado se o programa imediatamente anterior tiver terminado com uma codição cujo o código tiver sido maior que n. Se o comando está presente, mas nenhum valor é colocado, o valor de n será, por default, 1.

- Todos os programs terminam com uma código de condição de 16 se forem encontrados erros de setup. Por exemplo, se TABLES é executado imediatamente depois de TRANS, mas o usuário não quer executar TABLES se um erro de setup ocorreu na execução de TRANS, um comando $CHECK depois do comando $RUN TABLES previnirá a execução de TABLES.
- O comando $CHECK pode aparecer em qualquer lugar no setup do programa, mas é usualmente colocado imediatamente depois do comando $RUN.

$COMMENT [texto]. O "texto" desse comando é impresso na listagem do setup. Esse comando não tem nenhum efeito na execução do programa.

$DATA. O comando $DATA sinaliza que dados seguirão.

- Essa facilidade não pode ser usada se o programa gera um arquivo Dados de saída e um arquivo DATAOUT não foi especificado, i.e. os dados produzidos sejam gravados em um arquivo temporário default.
- Essa facilidade não pode ser usada se $MATRIX for usada.
- O comprimento de registro dos dados no setup não pode exceder 80 caracteres. Se registros ou linhas mais longos são introduzidas, apenas os primeiros 80 caracteres serão utilizados.
- O comando de impressão é desativado pelo comando $DATA. Portanto, a menos que um comando $PRINT siga imediatamente um comando $DATA, os dados não serão impressos.

$DICT. O comando $DICT sinaliza que um dicionário do IDAMS vem a seguir.

- Essa facilidade não pode ser usada se o programa gera um dicionário de saída e o arquivo DICTOUT não está especificado, i.e. o dicionário produzido ser gravado em um arquivo temporário default.
- O comando de impressão é desativado pelo comando $DICT. Portanto, a menos que um comando $PRINT siga imediatamente o comando $DICT, o dicionário não será impresso.

$FILES [RESET]. Sinaliza o início das especificações de arquivo. Nomes default de arquivos são colocados em cada arquivo no início da execução do(s) programa(s) do IDAMS através do uso de uma arquivo especial "idams.def". Qualquer um desses nomes default podem ser mudados pela introdução de declarações de especificação de arquivos depois do comando $FILES (ver "Especificações de Arquivo" abaixo). Para retornar aos nomes de arquivo default dos arquivos Fortran FT (exceto FT06 e FT50), usar o comando "FILES RESET".

$MATRIX. O comando $MATRIX sinaliza que uma matriz ou conjunto de matrizes virá.

- Essa facilidade não pode ser usada se a facilidade $DATA é usada.
- O comando de impressão é desativado pelo comando $MATRIX. Portanto, a menos que um comando $PRINT siga imediatamente o comando $MATRIX, a matriz utilizada não será impressa.

$PRINT. O comando de impressão é reverso; se estiver ativado, $PRINT o desativará; se estiver desativado, $PRINT o ativará. Quando o comando de impressão está ativado, as linhas do arquivo Setup são listadas como parte dos resultados do programa.

- Quando um comando $RUN é encontrado, o comando de impressão estará sempre ativado. Os comandos $DICT, $DATA, e $MATRIX automaticamente desativam o comando de impressão.

$RECODE. A ocorrência desse comando sinaliza que a faciliade Recode do IDAMS deve ser usada. A facilidade Recode é descrita no capítulo "Facilidade Recode" desse manual.

- As declarações Recode normalmente seguem um comando $RECODE. Se um comando novo do IDAMS segue imediatamente um comando $RECODE, declarações Recode do setup do programa anterior serão utilizadas.

$RUN programa. $RUN especifica o programa a ser executado e sempre é a primeira declaração no setup.

- "programa" é o nome do programa de 1 a 8 caracteres.
- Todos os comandos e declarações seguindo um comando $RUN e até o próximo comando $RUN se aplicam ao programa especificado.
- O comando de impressão é ativado quando um $RUN é encontrado. Ver a descrição do $PRINT.

$SETUP. O comando $SETUP marca o início das declarações de controle do programa, i.e. o filtro, título, declaração de parâmetros, etc. (ver abaixo).

- O comando $SETUP é requerido mesmo que declarações de controle sigam imediatamente o comando $RUN.

3.3 Especificações de Arquivo

Os nomes dos arquivos a serem utilizados são dados logo após o comando $FILES e possuem o seguinte formato:

ddname=filename [RECL=comprimento máximo do registro]

onde:

- ddname é o nome de referência usado internamente por programas, e.g. DICTIN. Os arquivos requeridos e os correspondentes ddnames para um programa particular são dados no write-up do programa na seção "Estrutura do Setup".
- filename é o nome do arquivo físico. Envolva o nome entre aspas se ele contém espaços em branco. Ver seção "Folders em WinIDAMS" para explicações adicionais.
- RECL deve ser usado caso o primeiro registro em um arquivo Dados não seja o maior. Se RECL não está especificado, o comprimento do registro é assumido ter o mesmo comprimento do primeiro registro. Se um registro subseqüente é maior, um erro de entrada ocorrerá.

Exemplos:

```
DATAIN = A:ECON.DAT RECL=92
PRINT  = RSLTS.LST
FT02   = ECON.MAT
DICTIN = \\nec0102\commondata\econ.dic
```

Para descrição mais aprofundada, ver seção "Customização do Ambiente para uma Aplicação" em capítulo "Interface do Usuário".

3.4 Exemplos do Uso de Comandos $ e Especificações de Arquivo

Exemplo A. Faz execuções múltiplas de um programa de análise, e.g. ONEWAY usando o mesmos dados mas com diferentes filtros, por exemplo.

```
$RUN ONEWAY
$FILES
DICTIN = CHEESE.DIC
DATAIN = CHEESE.DAT
$SETUP
     Filtro 1
```

```
     Outras declarações de controle para ONEWAY
$RUN ONEWAY
$SETUP
     Filter 2
     Outras declarações de controle para ONEWAY
```

Exemplo B. Executa TABLES e ONEWAY, usando o mesmo arquivo Dicionário e Dados para ambos e usando o mesmo Recode; não lista as declarações de Recode.

```
$RUN TABLES
$FILES
DICTIN = ABC.DIC
DATAIN = ABC.DAT  RECL=232
$SETUP
     Declarações de controle para TABLES
$RECODE
$PRINT
     Declarações de Recode
$RUN ONEWAY
$SETUP
     Declarações de controle para ONEWAY
$RECODE
$COMMENT DECLARAÇÕES DE RECODE PARA TABLES SERÃO REUTILIZADOS PARA ONEWAY
```

Exemplo C. Executa TABLES utilizando o Recode do IDAMS, dicionário no setup, dados em um disquete. Imprime o dicionário de entrada.

```
$RUN TABLES
$FILES
DATAIN = A:MYDATA
$RECODE
     Declarações de Recode
$SETUP
     Declarações de controle para TABLES
$DICT
$PRINT
     Dicionário
```

Exemplo D. Usa dados de um programa de gerenciamento de dados como entrada para progamas de análise sem reter o arquivo de saída, e.g. executa TRANS seguido de TABLES que utiliza os dados produzidos por TRANS através da especificação do parâmetro INFILE=OUT. TABLES não é executado se TRANS contiver erros de declaração.

```
$RUN TRANS
$FILES
DICTIN = MYDIC4
DATAIN = MYDAT4
$SETUP
     Declarações de controle para TRANS
$RECODE
     Declarações de Recode
$RUN TABLES
$CHECK
$SETUP
     Declarações de controle para TABLES incluindo o parâmetro INFILE=OUT
```

3.5 Declarações de Controle de Programa

3.5.1 Descrição Geral

As declarações de controle de programa do IDAMS (que seguem o comando $SETUP) são utilizadas para especificar os parâmetros para uma execução em particular. Existem três declarações de controle padrões usadas por todos os programas:

1. a declaração de filtro opcional para seleção de casos do arquivo de dados a ser usado,
2. a declaração de título mandatória que especifica um título para execução,
3. uma declaração de parâmetros mandatória que seleciona as opções para o programa; algumas opções são padrões em relação a maioria dos programas, outras são específicas de cada programa.

Declarações de controle de programa adicionais requeridas por programas individuais são descritas no write-up do programa.

3.5.2 Regras Gerais de Codificação

- Declarações de controle são introduzidas nas linhas com até 255 caracteres de comprimento.
- Linhas podem ser continuadas ao se digitar um traço no final da linha e continuando-se na outra linha.
- O comprimento máximo de informação que pode ser introduzido por uma declaração de controle é de 1024 caracteres excluindo-se a continuação de caracteres.
- Letras minúsculas, com exceção daquelas ocorrendo em strings entre aspas, são convertidas em maiúsculas.
- Se strings de caracteres entre aspas são incluídas em uma declaração de controle, elas devem ser continuadas em uma linha.

3.5.3 Filtros

Finalidade. Uma declaração de filtro é usada para selecionar um subconjunto dos casos dos dados. É expressa em termos de variáveis e valores assumidos por estas variáveis. Por exemplo, se a variável V5 indica, "sexo do respondente" em um survey e o código 1 representa feminino, então "INCLUDE V5=1" é uma declaração de filtro que especifica respondentes femininos como o subconjunto de casos desejado.

O filtro principal seleciona casos de um arquivo Dados de entrada e se aplica ao longo de toda a execução do programa. Esses filtros estão disponíveis em todos os programas do IDAMS que utilizam um dicionário (exceto BUILD e SORMER). Alguns programas permitem criar subconjuntos de dados adicionais. Essa filtragem "local" se aplica a apenas um programa específico, e.g. uma tabela de freqüência.

Exemplos.

```
1. INCLUDE V2=1-5 AND V7=23,27,35 AND V8=1,2,3,6
2. EXCLUDE V10=2-3,6,8-9 AND V30=<5 OR -
      V91=25
3. INCLUDE V50='FRAN','UK','MORO','INDI'
```

Localização. Se um filtro principal é usado, ele será sempre a primeira declaração de controle do programa. Cada write-up de programa indica se filtros "locais" também podem ser utilizados.

Regras de codificação.

- A declaração de filtro começa com a palavra INCLUDE ou EXCLUDE. Dependendo de qual palavra for dada, a declaração de filtro define o subconjunto de casos a ser utilizado pelo programa (INCLUDE) ou o subconjunto de casos a ser ignorado (EXCLUDE).

- Uma declaração pode conter um máximo de 15 expressões. Uma expressão consiste de um número de variável, um sinal de igual, e uma lista de valores possíveis. A lista de valores pode conter valores individuais e/ou intervalos de valores separados por vírgulas, e.g. V2=1,5-9. Intervalos abertos são indicados por < ou >, e.g INCLUDE V1=0,3-5,>10; contudo a variável deve sempre ser seguida de um sinal =, e.g V1>0 deve ser expresso como V1=>0 e V1<0 como V1=<0.
- Expressões são conectadas pelas conjunções AND e OR.
  - AND indica que um valor para cada uma das séries de expressões conectadas por AND deve ser achado.
  - OR indica que um valor de pelo menos uma das séries de expressões conectadas por OR deve ser encontrado.
- Expressões conectadas por AND são avaliadas antes de expressões conectadas por OR. Por exemplo, "expression-1 OR expression-2 AND expression-3" é interpretado como "expression-1 OR (expression-2 AND expression-3)". Portanto, para que um caso esteja no subconjunto definido por estas expressões, ou um valor de expression-1 ocorre, ou um valor de ambas expression-2 e expression-3 ocorrem, ou um valor ocorre em cada uma das três expressões.
- Parêntese **não podem** ser usados em declarações de filtro para indicar precedência na avaliação de expressões.
- Variáveis podem aparecer em qualquer ordem e em mais de uma expressão. No entanto, note que "V1=1 OR V1=2" é equivalente à expressão "V1=1,2". Note também que "V1=1 AND V1=2" é uma condição impossível, poi nenhum caso pode assumir simultaneamente um '1' e um '2' como valor da variável V1.
- Uma declaração de filtro pode opcionalmente ser terminada por um asterisco.
- As variáveis em um filtro.
  - Variáveis de tipo de caracter numérico e alfabético podem ser usadas.
  - Variáveis-R **não são** permitidas em filtros principais. Elas **são** permitidas em análises específicas ou filtros locais. Note que a declaração REJECT em Recode pode ser usada para filtrar casos de variáveis-R.
- Os valores em um filtro para variáveis numéricas.
  - Valores numéricos podem ser inteiros ou decimais, positivos ou negativos, e.g. 1, 2.4, -10.
  - Valores são expressos de maneira única ou em intervalos e são separados por vírgulas, e.g. 1-5, 8, 12-13.
  - Para variáveis de filtro numérico, valores de variável no arquivo de dados são primeiro convertidos para modo binário real usando o número correto de casas decimais do dicionário e a comparação com o valor do filtro é feita numericamente. Note que isso significa que para uma variável com casas decimais, valores de filtro devem ser definidos com o ponto decimal na posição correta, e.g. V2=2.5-2.8.
  - Casos onde a variável de filtro possui um valor não-numérico são sempre excluídos da execução.
- Os valores em um filtro para variáveis alfabéticas.
  - Valores de 1-4 caracteres são expressos como strings de caracteres entre aspas simples, e.g. 'F'. Espaços em branco não precisam ser incluídos, i.e. espaços em branco serão adicionados no final.
  - Se a variável possui uma largura de campo maior que 4, somente os primeiros 4 caracteres dos dados são utilizados para a comparação com a variável de filtro.
  - Somente valores únicos, separados por vírgulas são permitidos; intervalos de strings de caracteres não podem ser usados.

Note. A primeira declaração que segue ao comando $SETUP é reconhecido como o filtro principal se ela se inicia por INCLUDE ou EXCLUDE. Se o primeiro caracter que não esteja em branco for qualquer outra coisa, a declaração é assumida ser um título.

3.5.4 Títulos

Finalidade. Uma declaração de título é usada para identificar os resultados da execução de um programa. Alguns programas do IDAMS imprimem essa título uma vez no início dos resultados, equanto outros usam-na para identificar cada página.

Exemplos.

```
1. TABELAS DOS DADOS DA ELEIÇÃO DE 1998 - JULHO, 2000
2. IMPRESSÃO DOS DADOS CORRETOS DO SURVEY A34
```

Localização. Uma declaração de título **é requerida** por todos os programas do IDAMS. O título é a primeira ou (se um filtro é usado) a segunda declaração de controle de programa. Mesmo que nenhum tipo de título seja desejado, ainda assim é necessário incluir uma linha em branco.

Regras de codificação.

- A declaração pode ser uma string com quaisquer caracteres de onde os primeiros 80 caracteres serão utilizados, i.e. se um título maior que 80 caracteres for utilizada, ela é truncada nos primeiros 80 caracteres.
- Se o título não estiver entre aspas simples, letras minúsculas serão convertidas em maiúsculas e espaços em branco serão reduzidos a apenas um espaço em branco.
- O título não deve começar com as palavras “INCLUDE” ou “EXCLUDE”.

3.5.5 Parâmetros

Finalidade. Todos os programas do IDAMS foram desenhados de maneira bastante geral, permitindo ao usuário selecionar de um conjunto com várias opções. Essas opções e valores são gerados por parâmetros e são fornecidos nas declarações de controle de programa, como “parâmetros”, “especificações de regressão”, “especificações de tabela”, etc. Parâmetros são especificados pelo usuário através de um formato de palavra-chave padrão com uma palavra em Inglês ou abreviação usada para identificar a opção.

Exemplos.

```
1. WRITE=CORR WEIGHT=V3, PRINT=(DICT, PAIR)
   (PEARSON - parâmetros)
2. DEPV=V5 METHOD=STEP VARS=(R3-R9,V30) WRITE=RESID
   (REGRESSN - parâmetros da regressão)
3. ROWV=(V3,V9,V10) COLV=(V4,V11,V19) CELLS=(FREQ,ROWPCT) STATS=(CHI,TAUA)
   (TABLES - descrição da tabela)
```

Localização. A declaração de parâmetro principal **é requerida** por todos os programas IDAMS e deve seguir a declaração de título. Se todas as configurações default forem escolhidas, uma linha com um único asterisco deve ser fornecida. Cada write-up do programa indica o tipo e conteúdo de quaisquer outras listas de parâmetro necessárias e indica suas posições relativas a outras declarações de controle de programa.

Apresentação dos parâmetros de palavras-chave nos write-ups do programa. Todos os write-ups possuem uma notação padrão nas seções que descrvem os parâmetros do programa que estão disponíveis. A notação básica é a seguinte:

- Uma barra indica que apenas um dentre dois itens mutuamente exclusivos pode ser escolhidos, e.g. SAMPLE/POPUL ou PRINT=CDICT/DICT.
- Uma vírgula indica que todos, alguns, ou nenhum dos itens podem ser escolhidos, e.g. STATS=(TAUA, TAUB, GAMMA).
- Quando vírgulas e barras são combinadas, apenas um (ou nenhum) dos itens de cada grupo separado por vírgulas e conectado por barras pode ser escolhido, e.g. PRINT=(CDICT/DICT, LONG/SHORT).
- Defaults, se houver, aparecem em negrito, e.g. METHOD=**STANDARD**/STEPWISE/DESCENDING. Um default é uma configuração de parâmetro que o programa assume se uma seleção explícita não é feita pelo usuário.

- Quando uma configuração de parâmetro é obrigatória e não existe default, as palavras "Não há default" são usadas.
- Palavras em maiúsculas são palavras-chave. Palavras ou frases em minúsculas indicam que o usuário deve trocar a palavra ou frase por um valor apropriado, MAXCASES=n, VARS=(lista de variáveis).

Tipos de palavras-chave. Há 5 tipos de palavras-chave usadas para a especificação de parâmetros.

1. Uma palavra-chave seguida de uma string de caracteres. Esse tipo de palavra-chave identifica um parâmetro consistindo de uma string de caracteres, e.g.

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.

   Um usuário poderia especificar:

   INFILE=IN2
   (os ddnames seriam DICTIN2 e DATAIN2)

2. Uma palavra-chave seguida de um ou mais números de variáveis, e.g.

   WEIGHT= número de variável
   O número da variável peso se os dados devem ser ponderados.

   VARS=(lista de variáveis)
   Use apenas as variáveis da lista; os números podem ser listados em qualquer ordem com ou sem a notação-V, i.e. VARS=(V1-V3) ou VARS=(1-3). Note que os write-ups do programa sempre indicam se as variáveis tipos R e V ou apenas variáveis tipo-V podem ser utilizadas.

   Um usuário poderia especificar:

   WEIGHT=V39
   (a variável do peso é V39)

   VARS=(32,1,10)
   (apenas as variáveis especificadas devem ser usadas)

3. Uma palavra-chave seguida de um ou mais valores numéricos, e.g.

   MAXCASES=n
   Apenas os primeiros n casos serão processados.

   IDLOC=(s1,e1,s2,e2, ...)
   Colunas inicial e final de 1-5 campos de identificação de caso.

   Um usuário poderia especificar:

   MAXCASES=100
   (apenas os primeiros 100 casos serão usados)

   IDLOC=(1,3,7,9)
   (a ID do caso está localizada nas colunas 1-3 e 7-9)

4. Uma palavra-chave seguida por uma ou mais valores de palavra-chave. Os valores de palavras-chave podem ser uma mistura de opções mutuamente exclusivas (separadas por barras) e opções independentes (separadas por vírgulas). Por exemplo:

   PRINT=(**OUTDICT**/OUTCDICT/NOOUTDICT,DATA)

OUTD	Imprime o dicionário de saída sem registros-C.
OUTC	Imprime o dicionário de saída com os registros-C, se houver.
NOOU	Não imprime dicionário de saída.
DATA	Imprime os valores das variáveis de saída.

   Um usuário poderia especificar:

   PRINT=(OUTC,DATA)
   (o dicionário de saída completo é impresso, e os valores dos dados são impressos)

   PRINT=NOOUTDICT
   (nenhum dicionário de saída ou valores de dados são impressos)

5. Um conjunto de palavras-chave mutuamente exclusivas. Apenas uma de um conjunto de opções pode ser selecionada, e.g.

   SAMPLE/POPULATION

SAMP	Computa a variância e/ou desvio-padrão usando a equação da amostra.
POPU	Usa a equação da população.

Todas as palavras-chave exceto as do último tipo são seguidas por um sinal de igualdade. Os valores do caracter, numérico, e da palavra-chave que seguem o sinal de igualdade são chamados de "valores associados".

Regras de codificação.

Regras para especificação de palavras-chave

- Apenas as primeiras quatro letras de uma palavra-chave ou de uma palavra-chave associada precisam ser especificadas, apesar de que a palavra-chave completa pode ser especificada. Portanto, "TRAN" é uma forma abreviada apropriada para a palavra-chave "TRANSVARS". Não há abreviações para palavras-chave com quatro letras ou menos.

Regras de especificação de valores associados

- Valor associado é uma lista de itens.
  - Os itens na lista são separados por vírgulas.
  - Se existirem dois ou mais itens, a lista deve vir entre parênteses.
  - Intervalos de variáveis ou de valores numéricos inteiros devem ser indicados por uma barra.
  - Intervalos de valores numéricos decimais não são permitidos.

  Por exemplo:

  ```
  R=(V2,3,5)
  PRIN=(DICT,DATA,STAT)
  MAXC=5
  TRAN=(V5,V10-V25,V32)
  IDLOC=(1,3,7,8)
  ```

- Valor associado é uma string de caracteres.
  - A string deve vir entre aspas simples se contiver qualquer caracter não-alfanumérico, e.g. FNAME='EDUCATION: WAVE 1'. Note que espaços em branco, pontos e vírgulas são caracteres não-alfanuméricos. Se estiver em dúvida, use aspas simples.
  - Duas aspas simples consecutivas (não aspas duplas) devem ser usadas para representar uma aspa simples, e.g, ANAME='KEVIN"S' (a aspa simples é deletada, quanda a string é lida).
  - É melhor não separar strings ao longo das linhas.

Regras para especificação de listas de palavras-chave

- Palavras-chave (com ou sem valores associados) são separadas umas das outras por uma vírgula ou por espaços em branco, e.g

  ```
  FNAME='FRED', TRAN=3    KAISER
  ```

- Listas de palavras-chave podem se extender ao longo de diversas linhas, mas nesse caso deve haver um traço (-) no final de cada linha indicando a continuação, e.g.

  ```
  FNAME='FRED' -
  TRAN=3 -
  KAISER
  ```

- Palavras-chave podem ser atribuídas em qualquer ordem. Se uma palavra-chave aparece mais de uma vez em uma lista, então o último valor encontrado será utilizado.
- Uma palavra-chave não pode ser dividida no final de uma linha.
- Cada lista de palavras-chave pode, opcionalmente, ser finalizada por um asterisco.
- Se todas as opções default forem escolhidas, uma linha com um asterisco único deve ser fornecido.

Detalhes dos parâmetros mais comuns não descritos completamente em cada write-up do programa.

1. **BADDATA.** Tratamento de valores de dados não-numéricos.

   BADDATA=**STOP**/SKIP/MD1/MD2

   Quando caracteres não-numéricos (incluindo espaços em brancos envoltos e campos todo em branco) são encontrados em variáveis numéricas, o programa deveria:

STOP	Terminar a execução.
SKIP	Saltar o caso.
MD1	Trocar os valores não-numéricos pelo primeiro código de dados perdidos (ou 1.5×10^9 se o primeiro código de dados perdidos não for especificado).
MD2	Trocar os valores não-numéricos pelo segundo código de dados perdidos (ou 1.6×10^9 se o segundo código de dados perdidos não for especificado).

   Para SKIP, MD1, and MD2 uma mensagem é impressa informando o número de casos tratados.

2. **MAXCASES.** O número máximo de casos a ser processado.

   MAXCASES=n

   O valor dado é o número máximo de casos que será processado. Se n=0, nenhum caso será lido; essa opção pode ser usada para testar setups sem precisar ler os dados. Se o parâmetro não for especificado, todos os casos do arquivo de entrada serão lidos.

3. **MDVALUES.** Especifica quais dos códigos de dados perdidos que devem ser utilizados para checar a existência de dados perdidos nos valores das variáveis. Note que alguns programas possuem, em adição, um parâmetro MDHANDLING para especificar como os valores dos dados que estão perdidos devem ser tratados.

   MDVALUES=**BOTH**/MD1/MD2/NONE

BOTH	Os valores das variáveis são checados com os códigos do MD1 e com os intervalos de códigos definidos por MD2.
MD1	Valores de variáveis serão checados apenas em relação aos códigos do MD1.
MD2	Valores de variáveis serão checados apenas em relação aos intervalos de códigos definidos por MD2.
NONE	Os códigos MD não serão usados. Todos os valores serão considerados válidos.

   O default é que ambos os códigos MD são utilizados.

4. **INFILE, OUTFILE.** Especificando ddnames com os quais os arquivos de dicionários e de dados de entrada e saída são definidos.

   INFILE=**IN**/xxxx
   OUTFILE=**OUT**/yyyy

   Arquivos Dicionário e Dados para os programas do IDAMS são definidos com ddnames DICTxxxx, DATAxxxx, DICTyyyy e DATAyyyy. Se vários programas IDAMS são executados em um setup, por exemplo programas usando diferentes datasets como entrada, ou ao usar-se a saída de um programa como entrada diretamente em outro (encadeamento), então às vezes é necessário mudar esses defaults.

5. **WEIGHT.** Esse parâmetro especifica a variável cujos valores devem ser utilizados para ponderar os casos dos dados.

   WEIGHT=número de variável

   A variável especificada pode ser uma tipo-V ou tipo-R, inteira ou decimal. Casos com valores de peso perdidos, zero, negativo e não-numérico são sempre pulados e uma messagem é impressa a respeito dessas ocorrências. Se o parâmetro WEIGHT não é especificado, nenhuma ponderação é executada.

6. **VARS.** Esse parâmetro e outros similares como ROWVARS, OUTVARS, CONVARS, etc. são utilizados para especificar uma lista de variáveis.

   VARS=(lista de variáveis)
   Se mais de uma variável é especificada, a lista deve vir entre parênteses.

   Regras de especificação de listas de variáveis

   - Variáveis são especificadas por um "número" precedido por um V ou um R. Um V denota uma variável de um dataset IDAMS ou matriz. Um R denota uma variável resultante de uma operação Recode. Note que interno aos programas e nos resultados, variáveis tipo-V e tipo-R são diferenciadas pelo signal do número da variável; números positivos denotam variáveis do tipo-V e números negativos denotam variáveis denotam variáveis do tipo-R.
   - Para especificar um conjunto de variáveis numeradas continuamente, como V3, V4, V5, V6, conecte dois números de variáveis, cada uma precedida por um V, com um traço (e.g. V3-V6 é válido; V3-6 é inválido). Use intervalos com cuidado se a base de dados contiver vazios na numeração da variável, pois todas as variáveis no intervalo devem aparecer na base de dados ou matriz, i.e. V6-V8 implica V6,V7,V8. Se V7 não está no dicionário, então uma menssagem de erro aparecerá. Variáveis tipo-V e tipo-R não podem ser misturados em um intervalo, i.e. V2-R5 é inválido.
   - Números de variáveis únicas ou intervalos de números de variáveis são separadas por vírgula.
   - Em geral, para programas de gerenciamento de dados, variáveis podem ser listadas mais de uma vez, enquanto que para programas de análise, especificando mais de uma vez uma variável causará interrupção. Ver o write-up do programa para detalhes.
   - Espaços em branco podem ser inseridos em qualquer lugar na lista.
   - Em geral, variáveis podem ser especificadas em qualquer ordem. A ordem das variáveis pode, contudo, ter um significado especial em alguns programas; checar o write-up do programa para detalhes.

   Exemplos:

```
VARS=(V1-V6, V9, V16, V20-V102, V18, V11, V209)
OUTVARS=(R104, V7, V10-V12, R100-R103,  -
     V16, V1)
CONVARS=V10
```

3.6 Declarações de Recode

A facilidade Recode do IDAMS permite a recodificação temporária dos dados durante execução de programas do IDAMS. Resultados de tais operações de recodificação (juntamente com variáveis transferidas do arquivo de entrada) podem também ser salvos em arquivos permanentes usando o programa TRANS.

Recodificação é invocada pelo comando $RECODE. Esse comando e as declarações Recode associadas são colocadas depois do comando $RUN do programa onde será utilizada a facilidade Recode. Por exemplo:

```
$RUN programa                          $RUN ONEWAY
$FILES                                 $FILES
Especificações de arquivo              DICTIN=MYDIC
                                       DATAIN=MYDAT
$RECODE                                $RECODE
Declarações de Recode                   R10 = BRAC(V3,0-10=1,11-20=2)
                                        R11 = SUM(V7,V8)
                                        NAME R10 'EDUC LEVEL', R11'TOTAL INCOME'
$SETUP                                 $SETUP
Declarações de controle de programa    INCOME BY EDUC,SEX
                                       BADDATA=SKIP
                                       CONVARS=(R10,V2) DEPVAR=R11
```

Uma descrição completa da facilidade Recode é propiciada no capítulo "Facilidade Recode".

Capítulo 4

Facilidade Recode

4.1 Regras de Codificação

- Declarações de Recode possuem a seguinte forma:

  lab declaração

  onde lab é label de 1-4 caracteres opcional iniciando na posição 1 da linha e seguida por pelo menos um espaço vazio. Declarações sem label devem começar da posição 2 em diante.

- A label permite que declarações de controle como GO TO se refira a declarações específicas, e.g. GO TO ST1. Labels não podem ser definidas em declarações de inicialização (CARRY, MDCODES, NAME).

- Para continuar uma declaração em outra linha, entre com um traço no final da linha e continue a partir de qualquer posição na próxima linha.

- O comprimento máximo de linha é de 255 caracteres e o número máximo de caracteres para uma declaração é 1024 excluindo traços de continuação e espaços em branco no final, depois do traço.

4.2 Conjunto de Amostra de Declarações de Recode

Para dar uma idéia de como os elementos da liguagem Recode funcionam, um conjunto de amostras de declarações de Recode é dado abaixo.

```
$RECODE
   IF V5 LT 8 THEN REJECT                    (exclue casos onde V5 < 8)
   IF NOT MDATA(V6) THEN R51=TRUNC(V6/4) -
      ELSE R51=0
   R52=BRAC(V10,0-24=1,25-49=2,50-74=3, -    (agrupa valores de V10)
      74-99=4,TAB=1)
   R53=BRAC(V11,TAB=1)                       (agrupa V11 da mesmo jeito que V10)
   IF V26 INLIST(1-10) THEN R54=1 AND -
      R55=1 ELSE R54=2
   IF R54 EQ 1 THEN GO TO L1
   R55=99
   R56=V15 + V35
   GO TO L2
L1 R56=99
L2 R57=COUNT(1,V20-V27,V29)                  (conta quantas das variáveis listadas
                                              possuem o valor 1)
   NAME R52 'GROUPED AGE',  -
        R53 'GROUPED AGE AT MARRIAGE'
   MDCODES R55(99),R56 (99)
```

4.3 Manuseio de Dados Perdidos

Exceto em funções especiais MAX, MEAN, MIN, STD, SUM, VAR, Recode não checa automaticamente os valores das variáveis em relação a dados perdidos. O usuário deve, portanto, controlar especificamente os dados perdidos antes de realizar os cálculos com variáveis. A função MDATA está disponível para esse propósito; e.g.

```
IF MDATA (V5,V6) THEN R1=999 ELSE R1=V5+V6
```

Há duas funções adicionais, MD1 e MD2, que retornam o 1° ou 2° valor de código de dados perdidos para uma variável; e.g.

```
R2=MD1(V6)
```

aloca o valor do 1° código de dados perdidos de V6 em R2.

Finalmente, códigos de dados perdidos podem ser alocados a variáveis R ou V com a declaração de definição MDCODES; e.g.

```
MDCODES R3(8,9)
```

assinala 8 e 9 como o 1° e 2° códigos de dados perdidos para R3.

As vezes um conjunto de declarações de Recode não assinala valor algum para uma variável-R em um registro de dados particular. A variável-R assumirá o valor default de MD1, 1.5×10^9, para o qual é inicializado. Para mudar isso para um valor de dados perdidos mais aceitável, nós devemos testar se o valor é grande, caso positivo, assinale um valor de dado perdido apropriado, e.g.

```
IF R100 GT 1000000 THEN R100=99
MDCODES R100(99)
```

4.4 Como Recode Funciona

Checagem de sintaxe e interpretação. Declarações de Recode são lidas e analisadas em busca de erros antes da interpretação de outras declarações de controle de programa do IDAMS e antes da execução do programa. Se são encontrados erros, messagens de diagnóstico são impressas e a execução do programa é terminada.

Resultados. Recode imprime as declarações de Recode inseridas pelo usuário juntamente com os erros de sintax detectados, se houver. Isso ocorre antes do programa ser executado, i.e. antes da interpretação das declarações de controle do programa ser impressa.

Initialização antes de começar a processar o arquivo Dados. Se não houver erros de sintax, tabelas, códigos de dados perdidos, nomes, etc. são inicializados (de acordo com as declarações de inicialização/definição fornecidos pelo usuário) antes de iniciar a leitura dos dados. Variáveis-R em declarações CARRY são inicializadas em zero.

Initialização antes de processar cada caso de dados. No início do processamento de cada caso e antes da execução das declarações de Recode para aquele caso, todas as variáveis-R, exceto aquelas listadas nas declarações CARRY, são inicializadas no valor de dados perdidos de default, (1.5×10^9).

Execução de declarações de Recode. A recodificação acontece de fato depois que os dados de um caso são lidos e depois que o filtro principal foi aplicado. Casos que não atravessam o filtro não são passados para rotinas de recodificação. Variáveis Recode não podem, portanto, ser utilizadas em filtros principais.

O uso de declarações de Recode é sequêncial (i.e. a primeira declaração é usada primeiro, depois a segunda, terceira, etc.) exceto se for modificado por declarações GO TO, BRANCH, RETURN, REJECT, ENDFILE, ERROR (declarações de controle). Quando todas as declarações tiverem sido utilizadas, o caso é passado para o programa do IDAMS que está sendo executado.

Quando o programa do IDAMS tiver terminado de usar o caso, o caso seguinte passando o filtro é processado, as variáveis-R (exceto as variáveis CARRY) reinicializadas com os valores de dados perdidos e as declarações de Recode executados para aquele caso, até o final do arquivo de dados ser alcançado.

Testando declarações de Recode. Erros de lógica podem ser cometidos e não detectados pela facilidade Recode. Para checar os resultados esperados com aqueles gerados pelo Recode, as declarações de Recode devem ser testadas em alguns registros usando o programa LIST com o parâmetro MAXCASES tomando o valor de 10. Os valores de dados das variáveis usadas e as correspondentes variáveis resultantes podem então ser inspecionadas.

Arquivos usados pelo Recode. Quando um comando $RECODE é encontrado no arquivo Setup, linhas subsequentes são copiadas em um arquivo de trabalho na unidade FT46. O programa RECODE lê declarações de Recode desse arquivo e as analisa em busca de erros antes da execução do programa. Se são encontrados erros, mensagens de diagnóstico são impressas e a execução completa do IDAMS é terminada.

Declarações interpretadas são escritas na forma de tabelas em um arquivo de trabalho na unidade FT49, de onde são lidos pelo programa IDAMS sendo executado.

Mensagens em relação a declarações Recode são escritas na unidade FT06 com resultados do IDAMS sendo executados.

4.5 Operandos Básicos

Variáveis. Variáveis em Recode se referem ou a variáveis de entrada (Variáveis-V) ou a variáveis de resultado (Variáveis-R). Elas são definidas a seguir:

> Variáveis de entrada (Vn). Um "V" seguido por um número. Essas são variáveis definidas pelo dicionário de entrada. Seus valores podem ser modificados por Recode (e.g. V10=V10+V11). Variáveis devem ser normalmente numéricas, mas variáveis alfabéticas com não mais que 4 caracteres podem também ser utilizadas, em particular, elas podem ser recodificadas com valores numéricos.
>
> Variáveis de resultado (Rn). Um "R" seguido de um número (1 to 9999). São variáveis criadas pelo usuário. Variáveis-R (exceto aquelas listadas por declarações CARRY - ver abaixo) são inicializadas com o valor de dados perdidos default de 1.5×10^9 antes do processamento de qualquer caso.
>
> Para utilizar variáveis-R em um programa, especifique um R (ao invés de um V) na lista de variáveis anexada ao parâmetro de palavra-chave (e.g. WEIGHT=R50) ou VARS=(R10-R20)). Ao ser impresso pelo programa, um valor de variável de resultado é, às vezes, identificado por um sinal negativo. Portanto, variável "10" é R10 e variável "-10" é R10. É menos confuso utilizar números para as variáveis de resultado que sejam diferentes das variáveis de entrada. Variáveis-R são sempre numéricas.

Constantes numéricas. Constantes podem ser inteiras ou decimais, positivas ou negativas, e.g. (3, 5.5, -50, -0.5).

Constantes de caracteres. Constantes de caracteres vêm entre aspas simples (e.g. 'ABCXYZ', 'M'). Uma aspa simples ao longo de uma constante de caracter deve ser representada por duas aspas simples adjacentes (e.g. DON'T would be written: 'DON''T'). Constantes de caracteres são usadas na declaração NAME para definir nomes para novas variáveis. Elas também podem ser utilizadas em expressões lógicas para testar valores de variáveis alfabéticas (e.g. IF V10 EQ 'M'); apenas os primeiros 4 caracteres são usadas em tais comparações e constantes/variáveis de comprimento < 4 são completadas a direita com espaços em branco. Constantes de caracteres não podem ser utilizadas em funções aritméticas (exceto BRAC).

4.6 Operadores Básicos

Operadores aritméticos. Operadores aritméticos são usados entre operandos aritméticos. Operadores disponíveis, em ordem de precedência, são:

-	(negação)
EXP x	(exponenciação à potência x, onde -181 < x < 175)
*	(multiplicação)
/	(divisão)
+	(adição)
-	(subtração)

Operadores relacionais. Operadores relacionais são utilizados para determinar se dois valores aritméticos possuem, ou não, uma relação particular enter si. Os operadores relacionais são:

LT	(menor que)
LE	(menor que ou igual)
GT	(maior que)
GE	(maior que ou igual)
EQ	(igual)
NE	(diferente)

Operadores lógicos. Operadores lógicos são usados entre operandos lógicos. Operandos lógicos assumem apenas os valores de "verdadeiro" ou "falso". Eles são:

NOT	
AND	(ambos)
OR	(um ou outro)

4.7 Expressões

Uma expressão é uma representação de um valor. Uma constante simples, variável, ou referência à função é uma expressão. Combinação de constantes, variáveis, funções e outras expressões com operadores também são expressões. Recode pode avaliar expressões aritmética e lógicas. Note que parênteses podem ser utilizados em qualquer lugar na expressão para clarificar a ordem na qual deve ser avaliada.

Expressões aritméticas. Expressões aritméticas são criadas utilizando-se operadores aritméticos e variáveis, constantes, e funções aritméticas. Eles produzem um valor numérico. Exemplos são:

```
V732              (o valor de V732)
44                (a constante 44)
R67/V807 + 25     (25 mais o valor de R67 dividido pelo valor de V807)
LOG(R10)          (o log do valor de R10)
```

Expressões lógicas. Expressões lógicas são avaliadas como de valor "verdadeiro" ou "falso". Variáveis lógicas não existem na linguagem Recode, portanto o resultado de expressões lógicas não podem ser assumidos por uma variável. Expressões lógicas só podem ser utilizadas em declarações IF. Exemplos são:

```
R5 EQ V333
```

Verdadeiro se o valor de R5 é igual ao valor de V333, e falso caso contrário.

```
(V62 GT 10) OR (R5 EQ V333)
```

Verdadeiro se pelo menos uma das expressões lógicas resultar em valor verdadeiro e falso se ambas resultarem em um valor falso.

```
MDATA(V10,R20) AND V9 GT 2
```

Verdadeiro se o valor de V10 ou o valor de R20 é um código de dados perdidos e o valor de V9 é maior que 2, falso caso contrário.

4.8 Funções Aritméticas

Todas as funções aritméticas retornam um único valor numérico. A lista de argumentos das funções pode ser listas simples posicionadas entre parênteses ou listas altamente estruturadas envolvendo tanto elementos de palavra-chave quanto elementos em posições específicas da lista. As funções disponíveis são:

Função	*Exemplo*	*Finalidade*
ABS	`ABS(R3)`	Valor absoluto
BRAC	`BRAC(V5,TAB=1,ELSE=9, -` `1-10=1,11-20=2)`	Agrupamento univariado
	`BRAC(V10,'F'=1,'M'=2)`	Recodificação alfabética
COMBINE	`COMBINE V1(2), V42(3)`	Combinação de 2 variáveis
COUNT	`COUNT(1,V20-V25)`	Contagem da ocorrência de um valor ao longo de um conjunto de variáveis
LOG	`LOG(V2)`	Logaritmo na base 10
MAX	`MAX(V10-V20)`	Valor máximo
MD1,MD2	`MD1(V3)`	Valor do código de valor perdido
MEAN	`MEAN(V5-V8,MIN=2)`	Valor da média
MIN	`MIN(V10-V20)`	Valor mínimo
NMISS	`NMISS(V3-V6)`	Número de valores de dados perdidos
NVALID	`NVALID(V3-V6)`	Número de valores de dados não-perdidos
RAND	`RAND(0)`	Número randômico
RECODE	`RECODE V7,V8,(1/1)(1/2)=1, -` `(2-3/3)=2, ELSE=0`	Recodificação multivariada
SELECT	`SELECT (BY=V10,FROM=R1-R5,9)`	Selecionando o valor de um conjunto de variáveis de acordo com uma variável índice
SQRT	`SQRT(V2)`	Raiz quadrada
STD	`STD(V20-V25,MIN=4)`	Desvio-padrão
SUM	`SUM(V6,V8,V9-V12,MIN=3)`	Soma de valores
TABLE	`TABLE(V5,V3,TAB=2,ELSE=9)`	Recodificação bivariada
TRUNC	`TRUNC(V26/3)`	Parte inteira do valor de um argumento
VAR	`VAR(V6,R5-R10,MIN=7)`	Variância

A sintaxe exata para cada função é dada abaixo.

ABS. A função ABS retorna um valor que é o valor absoluto do argumento passado à função.

Protótipo: ABS(arg)

Onde arg é uma qualquer expressão aritmética de onde o valor absoluto de ve ser calculado.

Exemplo:

```
R5=ABS(V5-V6)
```

BRAC. A função BRAC retorna o valor que é obtido quando empregam-se operações específicas (regras) em uma única variável.

Protótipo: BRAC(var [,TAB=i] [,ELSE=valor] [,regra1,...,regra n])

Onde:

- var é qualquer variável do tipo-R ou V cujos valores são testados.
- TAB=i numera o conjunto de regras e o ELSE associado que foi estabelecido nesse uso de BRAC (opcional), ou faz referência a um conjunto de regras estabelecidas em um uso prévio de BRAC. Note: A cláusula ELSE é considerada parte do conjunto de regras.
- ELSE=valor é usado quando o valor de var não puder ser encontrado nas regras dadas. Se ELSE=valor é omitido, ELSE=99 é assumido, i.e. BRAC sempre recodificará.
- regra1, regra2,..., regra n são o conjunto de regras definindo os valores a serem retornados, dependendo do valor de var. As regras são expressas na forma de: x=c, onde x define um ou mais códigos e c é o

valor a ser retornado quando o valor de var for igual ao(s) código(s) definidos por x. As regras possíveis (onde n é qualquer constante numérica ou de caracteres) são:

>m=c (se o valor de var for maior que m, retornar o valor c).

<m=c (se o valor de var for menor que m, retornar o valor c).

m=c (se o valor de var for igual ao valor de m, retornar o valor c).

m1-m2=c (se valor de var estiver no intervalo de m1 até m2, i.e. m1<=var<=m2, retornar o valor c).

- Podem ser definidas quantas regras sejam necessárias. Elas são avaliadas da esquerda para a direita, e a primeira a ser satisfeita será utilizada. Note que ">" e "<" são utilizados, não os operadores lógicos GT e LT.
- ELSE, TAB, e as regras podem ser especificadas em qualquer ordem.
- Intervalos de valores alfabéticos não são permitidos, e.g. 'A'-'C', não é permitido.

Exemplos:

```
R1=BRAC(V10,TAB=1,ELSE=9,1-10=1,11-20=2,<0=0)
```

O valor de R1 será 1 se a variável 10 está no intervalo 1 to 10, 2 se V10 está no intervalo 11 - 20, e 0 se V10 é menor que 0. Se V10 assume qualquer outro valor, e.g. -3, 10.5, 25, 0, então a cláusula ELSE é aplicada, e R1 será 9. Essas regras de "bracketing" são nomeadas de table 1, de maneira que podem ser reutilizadas, e.g.

```
R2=V1 + BRAC(V2, TAB=1) * 3
```

Nesse exemplo, V2 seria "bracketed" pelas mesmas regras utilizadas em V10, no exemplo anterior. R2 assumiria os valores de V1 + (o resultado do "bracketing" multiplicado por 3).

```
R100=BRAC(V10,'F'=1,'M'=2,ELSE=9)
```

Esse é um exemplo de recodificação de uma variável alfabética, que assume valores 'F' ou 'M', com valores numéricos 1 e 2.

COMBINE. A função COMBINE retorna um único valor para cada combinação de valores das variáveis que são utilizadas como argumentos. Essa função é normalmente usada em variáveis categóricas.

Protótipo: COMBINE var1(n1), var2(n2),...,varm(nm)

Onde:

- var1 to var m são variáveis-R e -V a combinar-se.
- n1 to nm são os códigos máximos +1 das respectivas variáveis.
- A lista de argumentos para a função COMBINE não está entre parênteses.
- Cada variável deve assumir apenas valores inteiros não-negativos.
- Os valores retornados são computados pela seguinte fórmula:
  V1 + (n1 * V2) + (n1 * n2 * V3) + (n1 * n2 * n3 * V4) etc.
  O usuário, no entanto, determinaria normalmente o resultado da função listando as combinações de valores em uma tabela como no primeiro exemplo abaixo.

Exemplos:

```
R1=COMBINE V6(2), R330(3)
```

Assuma que V6 tem dois códigos (0,1) representando mulher e homem, respectivamente, e R330 possui três códigos (0,1,2) representando respondentes jovens, de meia idade e velhos, a declaração combinará os códigos de V6 e R330 para dar uma única variável R1 como o seguinte:

```
V6     V330     R1

 0      0        0      Homem novo
 1      0        1      Mulher nova
 0      1        2      Homem de meia idade
 1      1        3      Mulher de meia idade
 0      2        4      Homem velho
 1      2        5      Mulher velha
```

Como V6 possui dois códigos, e R330 possui 3, R1 terá seis. No exemplo acima, se V6 tivesse códigos 1 e 2 ao invés de 0 e 1, o valor máximo deveria ser declarado como"3". Isso incluiria os valores de 0,1, e 2, embora o código zero nunca apareça. Para evitar esses códigos "extras", o usuário deveria primeiro recodificar tais variáveis para obter um conjunto contíguo de códigos começando de 0, e.g. BRAC(V6,1=0,2=1). BRAC(V6,1=0,2=1).

Restrições:

- Deve haver no máximo 13 variáveis.
- A função COMBINE não pode ser usada com outras funções na mesma declaração de apontamento.
- Deve-se tomar cuidado para se especificar precisamente os códigos máximos quando se utilizar a função COMBINE. Caso contrário, valores não-únicos serão gerados. Por exemplo, com "COMBINE V1(2), V2(4)" a função retornará o valor de 7 para o par de valores, V1=1 e V2=3, e retornará também o valor de 7 para o par de valores V1=3 e V2=2. Se valores de 3 pudessem existir para V1, então n1 deveria ser especificado como 4 (1 + código máximo).

COUNT. A função COUNT retorna o valor que é igual ao número de vezes que o valor da variável ou constante ocorre como valor de uma das variáveis da lista "varlist".

Protótipo: COUNT(val,varlist)

Onde:

- val é normalmente uma constante, mas pode ser também uma variável-R ou -V.
- varlist dá as variáveis-R ou -V cujos valores devem ser testados em relação a val.

Exemplos:

```
R3=COUNT(1,V20-V25)
```

Será designado um valor para R3 igual ao número de vezes que o valor 1 ocorre nas 6 variáveis V20-V25. Isso poderia ser utilizado por exemplo para contar o número de respostas "SIM" de um respondente em um conjunto de perguntas.

```
R5=COUNT(V1,V8-V10)
```

Será designado um valor para R5 igual ao número de vezes que o valor de V1 ocorre nas variáveis V8-V10.

LOG. A função LOG retorna um valor ponto-flutuante que é o logaritmo na base 10 do argumento passado à função.

Protótipo: LOG(arg)

Onde arg é qualquer expressão aritmética para a qual o log da base 10 deve ser calculado.

Exemplos:

```
R10=LOG(V30)
```

Nota: O logaritmo de qualquer número X na base B pode ser calculado diretamente pela seguinte transformação:

```
R1=LOG(X)/LOG(B)
```

Para o logaritmo natural (base e), isso se torna simplesmente: R1=2.302585 * LOG(X).

Portanto R1=2.302585 * LOG(V30) designará para R1 o logaritmo natural da variável 30.

MAX. A função MAX retorna o valor máximo de um conjunto de variáveis. Valores de dados perdidos são excluídos. O argumento MIN pode ser usado para especificar o número mínimo de valores válidos para um máximo que deve ser calculado. De outra forma, o valor default de dados perdidos 1.5×10^9 é retornado.

Protótipo: MAX(varlist [,MIN=n])

Onde:

- varlist é uma lista de variáveis tipo-R e tipo-V, e constantes.
- n é o valor número mínimo de valores válidos para computação do valor máximo. n possui como valor default 1.

Exemplo:

```
R12=MAX(V20-V25)
```

MD1, MD2. A função MD1 (ou MD2) retorna o valor que é o primeiro (ou segundo) código de dados perdidos para a variável dada como argumento.

Protótipo: MD1(var) ou MD2(var)

Onde var é qualquer variável de entrada (variável-V) ou variável de resultado (variável-R) previamente definida.

Exemplo:

```
R12=MD2(V20)
```

Para cada caso processado, será designado para R12 o segundo código de dados perdidos da variável V20.

MEAN. A função MEAN retorna o valor da média de um conjunto de variáveis. Valores de dados perdidos são excluídos. O argumento MIN pode ser usado para especificar o número mínimo de valores válidos necessários para que a média seja calculada. Caso contrário, o valor default de dados perdidos de 1.5×10^9 é retornado.

Protótipo: MEAN(varlist [,MIN=n])

Onde:

- varlist é uma lista de variáveis-R e -V, e constantes.
- n é o número mínimo de valores válidos necessários para o cálculo do valor da média. O valor default de n é 1.

Exemplo:

```
R15=MEAN(R2-R4,V22,V5,MIN=2)
```

O resultado será a média das variáveis especificadas, se pelo menos duas delas tiver valores não-perdidos. Caso contrário, o resultado será 1.5×10^9.

MIN. A função MIN retorna o valor mínimo em um conjunto de variáveis. Valores de dados perdidos são excluídos. O argumento MIN pode ser usado para especificar o número mínimo de valores válidos para que o mínimo possa ser calculado. Caso contrário, o valor default de valores perdidos de 1.5×10^9 será retornado.

Protótipo: MIN(varlist [,MIN=n])

Onde:

- varlist é uma lista de variáveis tipo-R e -V, e constantes.
- n é o número mínimo de valores válidos para cálculo do valor mínimo. O valor default de n é 1.

Exemplo:

```
R10=MIN(V5,V7,V9,R2)
```

NMISS. A função NMISS retorna o número de valores perdidos em um conjunto de variáveis.

Protótipo: NMISS(varlist)

Onde varlist é uma lista de variáveis tipo-R e -V.

Exemplo:

```
R22=NMISS(R6-R10)
```

O valor retornado depende de quantas das variáveis R6 - R10 possuem valores perdidos. O valor máximo é 5 para um caso onde todas as 5 variáveis possuem dados perdidos.

NVALID. A função NVALID retorna o número de valores válidos (valores não-perdidos) em um conjunto de variáveis.

Protótipo: NVALID(varlist)

Onde varlist é uma lista de variáveis tipo-R e -V.

Exemplo:

```
R2=NVALID(V20,V22,V24)
```

O valor retornado depende de quantas das variáveis possuem valores válidos. O valor máximo de 3 será obtido se todas as variáveis possuem valores válidos. Será retornado 0 se todas as 3 são perdidas.

RAND. A função RAND retorna um valor que é um número aleatório distribuído uniformemente baseado nos argumentos 'starter" e "limit" como descrito abaixo:

Protótipo: RAND(starter [,limit])

Onde:

- starter é uma constante inteira que é utilizada para iniciar a sequência aleatória. Se o starter é 0, então o tempo do relógio corrente é utilizado.
- limit é um argumento opcional. É uma constante inteira que é utilizada para especificar o intervalo (i.e. 3 significa um intervalo de 1 a 3). O valor default é 10, o que significa que o intervalo default é 1 a 10.

Exemplos:

```
R1=RAND(0)
IF RAND(0) NE 1 THEN REJECT
```

Para cada caso processado, R1 assumirá um valor igual a um número randômico, uniformemente distribuído entre 1 e 10. A sequência é inicializada no tempo de relógio quando o RAND foi executado pelo primeira vez. Note que RAND pode ser usado com a declaração REJECT para selecionar uma amostra aleatória de casos. O 2° exemplo resultará na inclusão de 1/10 da amostra de casos, de maneira aleatória.

RECODE. A função RECODE é usada para retornar um valor baseado nos valores simultâneos de m variáveis.

Protótipo: RECODE var1,var2,...,varm [,TAB=i] [,ELSE=valor] [,regra1,regra2,...,regra n]

Onde:

- var1,var2,...,varm é uma lista de até 12 variáveis R e/ou V a serem testadas.
- TAB=i numera o conjunto de regras de recode estabelecidas nesse uso de RECODE (opcional) ou faz referência a um conjunto de valores estabelecidos em um uso prévio do RECODE. Note: o valor do ELSE **não é** considerado parte do conjunto de regras de recode.
- ELSE=valor (opcional) indica o valor a ser retornado se nenhuma das listas de código batem com os valores das variáveis. Mesmo sendo usualmente uma constante, o valor pode ser qualquer expressão

aritmética. Se ELSE é omitido e nenhuma das listas de código batem com os valores das variáveis, a função não retorna valor algum, i.e. o valor da variável de retorno é deixado como antes. Se esse for a primeira declaração de apontamento de uma variável, então o seu valor será o valor do dado de entradapara variável-V ou o valor de dados perdidos para uma variável-R.

- regra1, regra2,..., regra n é o conjunto de regras definindo os valores a serem retornados dependendo dos valores de var1, var2,..., varm. Cada regra tem a forma "(lista de códigos 1) (lista de códigos 2) ... (lista de códigos p)=c". Cada lista de códigos possui a forma "(a1/a2/.../am)" onde a1 é o código a ser comparado a var1, a2 é o código a ser comparado a var2, etc. Aqui, c é o valor a ser retornado quando var1,var2,..., varm batem com o código definido em quaisquer das listas de códigos.

  O protótipo para uma regra é:

  ```
  (a1/a2/.../am)(b1/b2/.../bm)...(x1/x2/.../xm)=c
  ```

  Cada lista de códigos contém uma lista e/ou um intervalo de valores para cada variável, e.g. com duas variáveis, (3/2)(6-9/4)(0/1,3,5)=1.
  Os códigos na lista de códigos podem ser separados por uma barra inclinada (indicando "AND") ou por uma barra vertical (indicando "OR"), apesar de que apenas uma ou outra pode ser utilizada numa dada lista de códigos.
  Por exemplo:

  ```
  (a1/a2/a3)=c (a função retornará c se var1=a1 e var2=a2 e var3=a3)
  ```

  ```
  (a1|a2|a3)=c (a função retornará c se var1=a1 ou var2=a2 ou var3=a3)
  ```

- Regras são examinadas da esquerda para a direita. A primeira lista de códigos que bater com a lista de variáveis determina o valor a ser retornado.
- A lista de argumentos para a função RECODE não vem entre parênteses.
- TAB, ELSE e regras podem estar em qualquer ordem.

Exemplos:

```
R7=RECODE V1,V2,(3/5)(7/8)=1,(6-9/1-6)=2
```

Será designado para R7 um valor baseado nos valores de V1 e V2. Nesse exemplo, R7 será igual a 1 se V1=3 e V2=5, ou se V1=7 e V2=8. R7 será igual a 2 se V1=6-9 e V2=1-6. Em qualquer outra situação, R7 permanecerá o mesmo (ver acima).

```
R7=RECODE V1,V2,TAB=1,ELSE=MD1(R7),(3/5)(7/8)=1,(6-9/1-6)=2
```

Será designado um valor igual ao do exemplo precedente, exceto que R7 será igual ao valor MD1 quando as regras não forem obedecidas. O TAB=1 permitirá que essas regras sejam utilizadas em outra chamada da função RECODE.

Restrição: Quando a função RECODE é usada, ela deve ser o único operando no lado direito do sinal de igual.

SELECT. A função SELECT retorna o valor da variável ou constante na lista FROM mantendo a mesma posição do valor da variável BY. (Atenção: Se o valor da variável BY for menor que 1 ou maior que o número de variáveis na lista FROM, um erro fatal ocorrerá). Podem haver até 50 itens na lista FROM. O valor máximo da variável BY é, portanto, 50. Uma função SELECT pode ser combinada com outras funções, operações, e variáveis para formar uma expressão complexa. Note: A **função** SELECT seleciona o **valor** de uma dentre um conjunto de variáveis; a **declaração** SELECT seleciona a **variável** a ser usada para o resultado. (Ver a seção especial "Declarações de Designação Especial" para descrição da declaração SELECT).

Protótipo: SELECT (FROM=lista de variáveis e/ou constantes, BY=variável)

Exemplo:

```
R10=SELECT (FROM=R1-R3,9,BY=V2)
```

R10 assumirá o valor de R1, R2, R3 or 9 para valores de 1, 2, 3 ou 4 respectivamente de V2.

SQRT. A função SQRT retorna o valor que é a raiz quadrada do argumento passado para a função.

Protótipo: SQRT(arg)

Onde arg é qualquer expressão aritmética.

Exemplo:

```
R5=SQRT(V5)
```

STD. A função STD retorna o desvio-padrão dos valores de um conjunto de variáveis. Valores perdidos são excluídos. O argumento MIN pode ser usado para especificar o número mínimo de valores válidos para que seja calculado o desvio-padrão. Caso contrário, o valor default de valor perdido de é retornado 1.5×10^9.

Protótipo: STD(varlist [,MIN=n])

Onde:

- varlist é uma lista de variáveis tipo-R e -V, e constantes.
- n é o número mínimo de valores válidos para que seja calculado o desvio-padrão. O valor default de n é 1.

Exemplo:

```
R5=STD(V20-V24,R56-R58,MIN=3)
```

SUM. A função SUM retorna a soma dos valores de um conjunto de variáveis. Valores perdidos são excluídos. O argumento MIN pode ser usado para especificar o número mínimo de valores válidos para que a soma possa ser calculada. Caso contrário, o valor default de valor perdido de é retornado 1.5×10^9.

Protótipo: SUM(varlist [,MIN=n])

Onde:

- varlist é uma lista de variáveis tipo-R, tipo -V e constantes.
- n é o número mínimo de valores válidos para que seja calculado a soma. O valor default de n é 1.

Exemplo:

```
R8=SUM(V20,V22,V24,V26,MIN=3)
```

Se três ou mais variáveis possuem valores válidos, a soma deles é retornada. Caso contrário, o valor de 1.5×10^9 é retornado.

TABLE. A função TABLE retorna o valor baseado nos valores simultâneos de duas variáveis.

Protótipo: TABLE (r, c, [TAB=i,] [ELSE=valor,] [PAD=valor,] COLS c1,c2,...,cm,

LINHA r1(valores da linha r1),r2(valores da linha r2),...,rn(valores da linha rn))

Onde:

- r é uma variável ou constante que será utilizada como um "índice de linha" para a tabela.
- c é uma variável ou constante que será utilizada como um "índice de coluna" para uma tabela.
- TAB=i numera a tabela definida nesse uso de TABLE (opcional) ou faz referência a tabela definida em um uso prévio de TABLE.
- ELSE=valor dá um valor para ser utilizado para pares de valores que não sejam definidos na tabela. O valor pode ser uma expressão aritmética. O valor de ELSE possui o valor default de 99 se não especificado, i.e. TABLE sempre retorna um valor.

- PAD=valor dá um valor para ser inserido em qualquer célula que é definida pelas especificações COLS mas não é definida pelas especificações de ROWS.
- TAB, ELSE e PAD podem ser especificadas em qualquer ordem.
- c1,c2,...,cm são as colunas para a tabela. Intervalos podem ser usados nas definições da coluna.
- r1,r2,...,rn são as linhas da tabela. O tamanho total da tabela será de m por n, onde m é o número de colunas e n é o número de linhas.
- (valores da linha r1), (valores da linha r2),...,(valores da linha rn) são os valores retornados dependendo dos valores r e c. Os valores são dados na mesma ordem das especificações da coluna; o primeiro valor corresponde a c1, o segundo a c2, etc. Intervalos podem ser usados nas definições dos valores de linha.

Exemplos: Assuma a seguinte tabela:

```
        Col:     1   2   3   4   5   6

Linha:  2        1   1   2   2   3   4
        3        1   2   2   2   3   4
        5        1   2   2   2   3   4
        6        3   3   3   3   3   4
        8        9   9   9   9   9   9
```

```
R1=TABLE (V6, V4, TAB=1, ELSE=0, PAD=9, COLS 1-6, ROWS 2(1,1,2,2,3,4), -
   3(1,2,2,2,3,4),5(1,2,2,2,3,4),6(3,3,3,3,3,4),8(9))
```

Se V6 é igual a 5 e V4 igual a 3, então o valor de 2 será designado para R1 (interseção da linha 5 com a coluna 3).
Se V6 é igual a 2 e V4 igual a 6, então o valor de 4 será designado para R1 (interseção da linha 2 com a coluna 6).
Se V6 é igual a 4 e V4 é igual a 2, então o valor de 0 será designado para R1 (linha 4 não está definida, o valor de ELSE é utilizado).

```
R5=TABLE (3, V8, TAB=7, ELSE=TABLE(V1,V8,TAB=1) )
```

Isto utilizará a tabela denominada "7" com 3 como o índice de linha e o valor de V8 como o índice de coluna. Se o valor de V8 não está na tabela 7 então a tabela "1" será usada com índice de linha V1 e índice de coluna V8.

TRUNC. A função TRUNC retorna o valor inteiro de um argumento.

Protótipo: TRUNC(arg)

Onde arg é qualquer expressão aritmética da qual o valor inteiro deve ser obtido.

Exemplo:

```
R5=TRUNC(V5)
```

R5 receberá o valor do entradada variável V5 truncada em um inteiro.

VAR. A função VAR retorna a variância dos valores de um conjunto de variáveis, excluindo-se valores perdidos. O argumento MIN pode ser usado para especificar o número mínimo de valores válidos para que a variância seja calculada. Caso contrário, o valor default de valores perdidos de 1.5×10^9 será retornado.

Protótipo: VAR(varlist [,MIN=n])

Onde:

- varlist é uma lista de variáveis do tipo-R e -V, e constantes.
- n é o número mínimo de valores válidos para o cômputo da variância. O valor default de n é 1.

Exemplo:

```
R9=VAR(V5-V10)
```

4.9 Funções Lógicas

Funções lógicas retornam um valor de "verdadeiro" ou "falso" quando avaliados. Elas não podem ser usadas como operandos aritméticos. Funções lógicas são utilizadas em expressões lógicas e expressões lógicas compõem a porção de testes de declarações condicionais "IF teste THEN...". As funções disponíveis são:

Função	*Exemplo*	*Finalidade*
EOF	`IF EOF THEN GO TO NEXT`	Checa o fim do arquivo de dados
INLIST	`IF V5 INLIST(2,4,6) THEN - R100=1 ELSE R100=0`	Faz busca em uma lista de valores
MDATA	`IF MDATA(V5,V6) THEN R101=99`	Checa a existência de valores perdidos

EOF. A função EOF é usada para agregação de valores ao longo dos casos. Ver exemplo 10 na seção "Exemplos do Uso de Recode". A presença da função EOF faz com que as declarações de Recode sejam executadas uma vez mais depois do fin do arquivo ser encontrado. O valor da função EOF é verdadeiro durante essa passagem depois-do-fim-do-arquivo da declaração de Recode e é falso em qualquer outro tempo.

Para a passagem final pelas declarações de Recode, variáveis-V terão os valores que tinham depois que o último caso foi completamente processado. Variáveis-R (exceto aquelas listadas em declarações CARRY) serão reinicializadas com o valor de 1.5×10^9. Variáveis-R listadas no CARRY não serão modificadas. O usuário deve ser cuidadoso em arrumar um atalho correto a ser pecorrido através de declarações de Recode quando o fim-do-arquivo é alcançado.

Protótipo: EOF

Exemplo:

```
IF R1 NE V1 OR EOF THEN GO TO L1
```

INLIST. A função INLIST (abreviada por IN) retorna o valor de "verdadeiro" se o resultado de uma expressão aritmética pertence a um conjunto especificado de valores. Se a expressão é igual a um valor fora do conjunto de valores, a função retorna o valor de "falso".

Protótipo: expr INLIST(valores) ou expr IN(valores)

Onde:

- expr é qualquer expressão aritmética ou uma única variável.
- valores é uma lista de valores. Eles podem ser discretos e/ou um intervalo de valores.

Exemplos:

```
IF R12 INLIST(1-5,9,10) THEN V5=0
```

Se R12 tiver um valor de 1,2,3,4,5,9 ou 10, a função INLIST retorna o valor de "verdadeiro", a variável de entrada V5 é igualada a 0. Caso contrário, INLIST retorna o valor de "falso" e a variável de entrada mantém seu valor original.

```
IF (V3 + V7) IN(2,4,5,6) THEN R1=1 ELSE R1=9
```

Se a soma das variáveis de entrada V3 e V7 resulta no valor 2,4,5, ou 6, então INLIST retorna um valor de "verdadeiro" e a variável de resultado assumirá o valor de 1. Caso contrário, INLIST retorna um valor de "falso" e R1 será igualado a 9.

MDATA. A função MDATA retorna um valor de "verdadeiro" se qualquer das variáveis passadas para a função possui valores de dados perdidos; caso contrário, a função retorna um valor de "falso". Essa função é usada com frequência, dado que valores perdidos não são automaticamente checados na avaliação de expressões, exceto em funções MAX, MEAN, MIN, STD, SUM e VAR.

Protótipo: MDATA(varlist)

Onde varlist é uma lista de variáveis do tipo-R e -V. Pode haver um máximo de 50 variáveis nessa lista.

Exemplo:

```
IF MDATA(V1,V5-V6) THEN R1=MD1(R1) ELSE R1=V1+V5+V6
```

Se qualquer variável na lista V1, V5, V6 tem um valor igual a seu código MD1 ou no intervalo especificado pelo código MD2, a função MDATA retornará um valor de "verdadeiro", e a variável de resultado R1 será igualada ao primeiro código de dados perdidos. Caso contrário, a função MDATA retornará o valor de "falso" e R1 é igualado a soma de V1, V5, V6.

4.10 Declarações de Designação

Estas são as principais unidades estruturais da linguagem Recode. Elas são utilizadas para designar um valor para um resultado. Qualquer número entre 1 e 9999 pode ser usado por uma variável-R mas pode-se evitar confusão se os números-R forem distintos dos números-V das variáveis no dicionário de entrada, e.g. se existirem 22 variáveis no dicionário então comece a numerar as variáveis-R de R30. Declarações de designação podem também ser utilizadas para designar um novo valor para uma variável de entrada. Nesse caso o valor original da variável de entrada é perdido durante a duração da execução do programa IDAMS.

Protótipo: variável=expressão

Onde:

- variável é qualquer variável de entrada (Vn) ou de resultado (Rn).
- expressão é qualquer expressão aritmética usando, como opção, funções aritméticas de Recode.
- Note que as variáveis usadas na expressão não são automaticamente checadas em relação a valores perdidos, exceto quando se trata das funções especiais MAX, MEAN, MIN, STD, SUM, VAR. Em todos os outros casos, declarações específicas para checar dados perdidos devem ser introduzidas no lugar apropriado. Ver abaixo em "Declarações condicionais" por exemplo.

Exemplos:

```
R10=5
```

Para o valor de R10 é designada a constante 5.

```
R5=2*V10 + (V11 + V12)/2
```

Qualquer expressão aritmética pode ser usada e parênteses são utilizados para modificar a precedência dos operadores aritméticos.

```
V20=SQRT(V20)
```

O valor de V20 é trocado pela valor de sua raiz ao quadrado usando a função SQRT.

```
R20=BRAC(V6,0-15=1,16-25=2,26-35=3,36-90=4,ELSE=9)
```

Os valores de 1, 2, 3, 4 ou 9 são designados para R20 de acordo a que grupo o valor de V6 pertence.

```
R10=MD1(V10)
```

O primeiro código de dados perdidos de V10 é designado como R10.

4.11 Declarações Especiais de Designação

DUMMY. A declaração DUMMY produz uma série de "variáveis dummies", codificadas 0 ou 1, a partir de uma única variável.

Protótipo: DUMMY var1,...,varn USING var(val1)(val2)...(valn)[ELSE expressão]

Onde:

- var1, var2,...,varn é uma lista de variáveis dummy cujos valores são definidos por esta declaração. Elas podem ser variáveis-R ou -V, podem ser listadas como valores únicos ou intervalos, e devem vir separadas por vírgulas (e.g. R1-R3, R10, R7-R9, V20). A ordem especificada é preservada.

- Referências dobradas (R1, R3, R1) são válidas.

- var é qualquer variável-R ou -V. O valor dessa variável é testado em relação às listas de valores (val1)(val2) etc. para calcular o valor apropriado das variáveis dummy.

- (val1)(val2)...(valn) são listas de valores usados para ajustar os valores das variáveis dummy. Deve haver o mesmo número de listas e de variáveis dummy (var1, var2, ...,varn). Listas de valores podem conter constantes únicas ou intervalos ou ambos.

- expressão é qualquer expressão aritmética que é usada como o valor para todas as variáveis dummy quando o valor da variável var não é um dos valores das listas. O valor default de expressão é a constante 0.

- O valor da variável var é testado em relação às listas de valores (o número de listas de valores deve ser igual ao número de variáveis dummy); se var possuir um valor na primeira lista de valores, a primeira variável dummy é igualada a 1, as outras a 0; se o valor de var ocorre na segunda lista de valores, a segunda variável dummy é igualada a 1, as outras a 0, etc. Se o valor de var não ocorre em nenhuma lista de valores, todas as variáveis dummy são igualadas ao valor especificado depois de ELSE (O default é 0).

Exemplo:

```
DUMMY R1-R3 USING V8(1-4)(5,7,9)(0,8) ELSE 99
```

O seguinte quadro mostra os valores de R1, R2 e R3 baseados em diferentes valores de V8:

V8:	1	2	3	4	5	7	8	9	0	OUTROS
R1:	1	1	1	1	0	0	0	0	0	99
R2:	0	0	0	0	1	1	0	1	0	99
R3:	0	0	0	0	0	0	1	0	1	99

SELECT. A declaração SELECT faz com que a **variável** na lista FROM ocupando a mesma posição descrita pelo valor da variável BY, seja igualada ao valor da expressão a direita do sinal de igualdade i.e. ele seleciona **qual** variável deve receber o valor. Se o valor da variável BY é menor que 1 ou maior que o número de variáveis na lista FROM, um erro fatal resultará. O número máximo de ítens na lista FROM é 50. Portanto, o valor máximo da variável BY é 50.

Protótipo: SELECT (FROM=lista de variáveis,BY=variável)=expressão

Exemplos:

```
SELECT (FROM=R1,V3-V10, BY=R99)=1
SELECT (BY=V1, FROM=V8,R2,R5)=R7*5
```

No primeiro exemplo, R1 será igual a 1 se R99 for igual a 1; V3 será igual a 1 se R99 igual a 2; ... ; e V10 será 1 se R99 igual a 9. Se R99 for maior que 9 ou menor que 1, um erro fatal ocorrerá. Os valores das oito variáveis não selecionadas não serão alterados.

SELECT pode ser usado para formar um loop da seguinte maneira:

```
    R99=1
L1  SELECT (BY=R99, FROM=R1,V3-V10)=0
    IF R99 LT 9 THEN R99=R99+1 AND GO TO L1
```

As nove variáveis R1, V3-V10 serão igualadas a zero, uma após a outra, a medida que R99 é incrementada de 1 a 9. O loop estará completo quando R99 igualar-se a 9 e todos as variáveis tenham sido inicializadas.

4.12 Declarações de Controle

Declarações de Recode são executadas normalmente em cada caso de dados na ordem do primeiro até o último. A ordem pode ser modificada com uma das seguintes declarações de controle:

Declaração	*Exemplo*	*Finalidade*
BRANCH	`BRANCH (V16,L1,L2)`	"Branch" dependendo do valor de uma variável
CONTINUE	`CONTINUE`	Continue com a próxima declaração
ENDFILE	`ENDFILE`	Não processe mais casos de dados após este
ERROR	`ERROR`	Termine execução completamente
GO TO	`GO TO TOWN`	"Branch" incondicionalmente
REJECT	`REJECT`	Rejeitar o caso de dados corrente
RELEASE	`RELEASE`	Liberar o caso de dados corrente para o programa para processamento e, então, executar declarações de recodificação novamente sem ler outro caso
RETURN	`RETURN`	Use o caso corrente para análise sem recodificação adicional

BRANCH. A declaração BRANCH muda a seqüência na qual as declarações são executadas, dependendo do valor de uma variável.

Protótipo: BRANCH(var,labels)

Onde:

- var é uma variável-R ou -V.
- labels é uma lista de um ou mais labels de declaração de caracteres de 1-4.

Exemplo:

```
BRANCH(R99,LAB1,LAB2,LAB3)
```

Transferência é feita para LAB1, LAB2, ou LAB3, dependendo se R99 possui um valor de 1,2, ou 3.

CONTINUE. CONTINUE é uma declaração simples que não executa operação alguma. É usada como um ponto conveniente de transferência.

Protótipo: CONTINUE

Exemplo:

```
        IF V17 EQ 10 THEN GO TO AT
        R10=V11
        GO TO THAT
AT      R20=V11*100
THAT    CONTINUE
```

ENDFILE. A declaração ENDFILE faz com que a facilidade Recode feche a base de dados de entrada exatamente como se um fim-de-arquivo fosse atingido. Se a função EOF tiver sido especificada, será designado para a função EOF um valor verdadeiro para uma passagem final através das declarações de Recode desde o começo, depois de ENDFILE tiver sido executado.

Protótipo: ENDFILE

Exemplo:

```
IF V1 EQ 100 THEN ENDFILE
```

Essa declaração pode ser usada para testar um conjunto de declarações Recode ou um setup do IDAMS nos primeiros n casos de um dataset.

ERROR. A declaração ERROR direciona a facilidade Recode a terminar a execução com uma mensagem de erro que indica o número de casos e o número de declaração de Recode na qual o erro ocorreu.

Protótipo: ERROR

Exemplo:

```
        IF R6 EQ 2 THEN GO TO B
        ERROR
B       CONTINUE
```

GO TO. A declaração GO TO é usada para mudar a sequência na qual as declarações são executadas. Na ausência de uma declaração GO TO ou BRANCH, cada declaração é executada sequêncialmente.

Protótipo: GO TO label

Onde label é uma label de declaração de caracteres 1-4. A declaração identificada pela label pode vir fisicamente antes ou depois da declaração GO TO. (Atenção: Seja cuidadoso ao referir-se a uma declaração antes de GO TO, pois um loop sem fim pode ser formulado.

Exemplo:

```
        GO TO TOWN
        .
        .
        R10=R5
        GO TO 1
TOWN    R10=R5+V11
1       R11=...
```

REJECT. A declaração REJECT direciona a facilidade Recode a rejeitar o caso presente e obter outro caso. O novo caso é então processado do começo das declarações de Recode. Portanto, REJECT pode ser utilizado como um filtro com as variáveis-R.

Protótipo: REJECT

Exemplo:

```
IF MDATA (V8,V12-V13) THEN REJECT
```

RELEASE. A declaração RELEASE direciona a facilidade Recode a liberar o caso presente para o processamento do programa e para reaver controle depois do processamento sem ler um outro caso. Depois de reaver o controle, Recode reassume com a primeira declaração de Recode. RELEASE pode ser utilizada para decompor um único registro em vários casos para análise. Note: Ao utilizar a declaração RELEASE, deve-se tomar cuidado para não se ter um processamento que continue indefinidamente.

Protótipo: RELEASE

Exemplo:

```
CARRY (R1)
R1=R1+1
IF R1 LT V1 THEN RELEASE ELSE R1=0
```

RETURN. A declaração RETURN direciona a facilidade Recode para retornar o controle para o programa IDAMS. Nem uma outra declaração do IDAMS é executado para o caso corrente.

Protótipo: RETURN

Exemplo:

```
        IF V8 LT 12 THEN GO TO A
        RETURN
A       R10=V8
```

4.13 Declarações Condicionais

A declaração **IF** permite a designação condicional e/ou controle condicional. É uma declaração composta onde várias declarações são conectadas pelas palavras-chave THEN, AND e ELSE.

Protótipo:

IF test THEN stmt1 [AND stmt2 AND ... stmt n][ELSE estmt1] [AND estmt2 AND ... estmt n]

Onde:

- test pode ser qualquer combinação de expressões lógicas (incluindo funções lógicas) conectadas por AND ou OR e precedidas opcionalmente por NOT. Pode ser, mas não necessita ser, colocada entre parênteses.
- stmt1,...,stmt n,estmt1,...,estmt n pode ser qualquer designação ou declaração de controle (exceto CONTINUE).
- A(s) declaração(ões) entre o THEN e ELSE são executadas se o teste é verdadeiro.
- A(s) declaração(ões) depois do THEN e ELSE são executadas se o teste é falso. Se nenhuma cláusula ELSE está presente, a próxima declaração é executada.
- As palavras-chave THEN e ELSE podem, cada uma, ser seguidas de qualquer número de declarações, cada uma conectada pela palavra-chave AND.

Exemplos:

```
IF V5 EQ V6 THEN R1=1 ELSE R1=2
```

Faz R1 igual a 1 se o valor de V5 igualar o de V6; caso contrário faz R1 igual a 2.

```
IF MDATA(V7,V10-V12) THEN R6=MD1(V7) AND R10=99 -
   ELSE R6=V7+V10+V11 AND R10=V12*V7
```

Iguala R6 ao primeiro valor de dados perdidos de V7 e R10 a 99 se quaisquer das variáveis V7, V10, V11, V12 forem iguais aos seus códigos de valores perdidos. Caso contrário, iguala R6 a soma de V7, V10 e V11, e também iguala R10 ao produto de V12 e V7.

```
IF (V5 NE 7 AND R8 EQ 9) THEN V3=1 ELSE V3=0
```

Faz V3 igual a 1 se, simultaneamente, V5 não é igual a 7 e R8 é igual a 9. (NOte: Os parênteses não são requeridos).

```
IF MDATA(V6) OR V10 LT 0 THEN GO TO X
```

Se o valor de V6 é perdido ou V10 é menor que 0, desvie para a próxima declaração marcada com X; caso contrário, continue com a próxima declaração.

4.14 Declarações de Inicialização/Definição

Essas declarações são executadas de uma vez, antes do processamento de dados começar, para inicializar valores que serão utilizados na execução das declarações de Recode. Elas não podem ser usadas em expressões e não podem ter labels.

CARRY. A declaração CARRY faz com que os valores das variáveis listadas sejam transportados de caso em caso. Variáveis CARRY são inicializadas apenas uma vez (antes de se iniciar a leitura de dados) com valor 0. As variáveis CARRY podem ser usadas como contadores ou como acumuladores para agregação.

Protótipo: CARRY(varlist)

Onde varlist é uma lista de variáveis-R.

Exemplo:

```
CARRY(R1,R5-R10,R12)
```

MDCODES. A declaração MDCODES muda de códigos de dados perdidos do dicionário para as variáveis de entrada ou estabelece códigos de dados perdidos para variáveis de resultado. Os default usados por Recode para variáveis-R e -V sem especificação de dados perdidos no dicionário e sem especificação de MDCODES são MD1=1.5×10^9 e MD2=1.6×10^9.

Protótipo: MDCODES (varlist1)(md1,md2),(varlist2)(md1,md2), ..., (varlistn)(md1,md2)

Onde:

- varlist1, varlist2, ..., varlistn são listas de variáveis contendo listas de variáveis únicas e intervalos de variáveis.
- md1 and md2 são o primeiro e segundo código de dados perdidos, respectivamente, para todas as variáveis listadas. Códigos de dados perdidos para valores decimais devem ser especificados com um ponto decimal explícito. Cuidado: apenas duas casas decimais são retidas pelas variáveis-R, arrendondando-se o valor, e.g. md1 especificado como 9.999 é tratado como 10.00.
- Ou md1 ou md2 deve ser omitida. Se md1 é omitida, uma vírgula deve preceder o valor de md2.

Exemplos:

```
MDCODES V5(8,9)
```

O primeiro código de dados perdidos para V5 será 8; o segundo código de dados perdidos será 9.

```
MDCODES (R9-R11)(,99), V7(8,9), V6(9)
```

Para R9, R10 e R11, o primeiro código de dados perdidos será 1.5×10^9 e o segundo código de dados perdidos será 99.
Para V7, o primeiro código de dados perdidos será 8 e o segundo código de dados perdidos será 9
Para V6, o primeiro código de dados perdidos será 9 e o segundo código de dados perdidos será 1.6×10^9.

NAME. A declaração NAME designa nomes para variáveis-R ou renomea variáveis-V.

Protótipo: NAME var1 'name1' ,var2 'name2', ..., varn 'name n'

Onde:

- var1,var2,...,varn são variáveis-R ou -V.
- name1, name2,...,name n são nomes para colocar nessas variáveis.
- O número máximo de caracteres por nome é 24; se for mais longo, o nome é truncado em 24 caracteres.
- O nome default para uma variável-R é 'RECODED VARIABLE Rn'.
- Para incluir um apóstrofe em um nome (e.g. PERSON'S), use duas aspas simples (e.g. PERSON"S).

Exemplo:

```
NAME R1 'V5 + V6', V1 'PERSON''S STATUS'
```

4.15 Exemplos do Uso de Declarações de Recode

Suponha que exista um arquivo de dados com as seguintes variáveis:

V1	ID da Vila	
V2	Sexo	1=masculino, 2=feminino
V4	Idade	21-98, 99=não declarado
V5	Nível de Educação	1=primário, 2=secundário, 3=universitário, 9=não declarado
V8	Renda do primeiro trabalho	
V9	Renda do segundo trabalho	
V10	Renda do parceiro	
V21	Peso em kg (uma decimal)	
V22	Altura em metros (2 decimais)	
V31	Possui carro?	1=sim, 2=não, 9=ND
V32	Possui TV?	
V33	Possui som stéreo?	
V34	Possui freezer?	
V35	Possui Microcomputador?	
V41	Número de crianças	
V42	Idade da primeira criança	
V43	Idade da segunda criança	
V44	Idade da terceira criança	
V45	Idade da quarta criança	

Maneiras de se construir algumas análises possíveis desses dados são evidenciadas abaixo.

1. Renda total. Se a renda do primeiro e do segundo trabalho estão ambas faltando, a renda total será um dado perdido. Se apenas uma observação de renda está faltando então use esse valor como o total.

```
          IF NVALID(V8,V9) EQ 0 THEN R101=-1 AND GO TO END
          IF NVALID(V8,V9) EQ 2 THEN R101=V8+V9 AND GO TO END
          IF MDATA(V8) THEN R101=V9 ELSE R101=V8
     END  CONTINUE
          MDCODES R101(-1)

ou        R101=SUM(V8,V9,MIN=1)
          IF R101 EQ 1.5 * 10 EXP 9 THEN R101=-1
          MDCODES R101(-1)
```

2. Não utilize o caso se a renda total é zero ou está faltando.

```
IF MDATA(R101) OR R101 EQ 0 THEN REJECT
```

3. Uma composição da renda tomando 3/4 da própria renda mais 1/4 da renda do cônjuge. Se a renda do cônjuge está faltando, assuma que é zero.

```
IF MDATA(V10) THEN V10=0
IF MDATA(R101) THEN R102=MD1(R102)  -
   ELSE R102=R101 * .75 + V10 * .25
NAME R102'Composite income'
MDCODES R102(99999)
```

4. Peso do respondente agrupado entre leve (30-50), médio (51-70) e pesado (70+).

```
R103=BRAC(V21,30-50=1,50-70=2,70-200=3,ELSE=9)
```

   Note que V21 é armazenada com uma casa decimal. Para se ter certeza que valores como 50.2 sejam designado para uma categoria, intervalos na declaração BRAC devem se sobrepor. Recode funciona da esquerda para a direita e designa o código para o primeiro intervalo dentro do qual o caso se localiza. Portanto, um valor de 50.0 cairá na categoria 1, mas um valor de 50.1 cairá na categoria 2. Para colocar valores de 50 na segunda categoria, use

```
R103=BRAC(V21, <50=1, <70=2, <200=3, ELSE=9)
```

Um valor de 49 seria classificado em todos os três intervalos, mas Recode usará o primeiro intervalo válido que encontrar (código 1). Um valor de 50 não satisfará o primeiro intervalo e será designdo para o intervalo 2.

5. Índice de riqueza com valores 0-5 de acordo com o número de bens possuídos.

```
R104=COUNT(1,V31-V35)
```

Se todos os ítens são codificados 1(sim), o índice R104 terá valor de 5. Se todos são codificados com 2 (não) ou estão faltando, então o índice será zero.

6. Cria 3 variáveis dummy (codificadas 0/1) da variável de educação.

```
DUMMY R105-R107 USING V5(1)(2)(3)
```

As três variáveis de resultado assumirão os seguintes resultdos:

V5=1	R105=1, R106=0, R107=0
V5=2	R105=0, R106=1, R107=0
V5=3	R105=0, R106=0, R107=1
V5 not 1,2 or 3	R105=0, R106=0, R107=0 (default se nenhum valor para ELSE for dado)

7. Idade da criança mais nova. Idades das últimas 4 crianças são armazenadas nas variáveis 42 a 45, a mais velha sendo a que está em V42. Se alguém possui 3 crianças, então o valor de V44 dá a idade da criança mais nova; se alguém tem 4 ou mais crianças então nós queremos V45. Nesse caso, V41 (número de crianças) pode ser usado como um índice para selecioinar a variável correta utilizando a função SELECT.

```
IF V41 GT 4 THEN V41=4
IF V41 EQ 0 OR MDATA(V41) THEN  R109=99 ELSE  -
   R109=SELECT (FROM=V42-V45, BY=V41)
NAME R109'Last child''s age'
MDCODES R109(99)
```

8. Quociente Peso/Altura como um número decimal e arredondado para o inteiro mais próximo.

```
IF MDATA (V21,V22) OR V22 EQ 0 THEN R111=99 AND R112=99 -
   ELSE R111=V21/V22 AND R112=TRUNC ((V21/V22) + .5)
NAME R111'Weight/Height ratio dec', R112 'W/H rounded'
MDCODES (R111,R112)(99)
```

9. Cria uma única variável combinando sexo e nível educacional em 4 grupos como o seguinte:

Mulheres, educação primária apenas
Mulheres, educação secundária+
Homems, educação primária apenas
Homems, educação secundária+

Método a. Primeiro reduza os códigos de sexo e educação a códigos adjacentes começando de 0, armazenando temporariamente os resultados nas variáveis R901, R902.

```
R901=BRAC (V5,1=0,2=1,ELSE=9)
R902=BRAC (V6,1=0,2=1,3=1,ELSE=9)
```

Então, use a função COMBINE, tendo certeza de que os casos com códigos espúrios são colocados na categoria de dados perdidos.

```
IF R901 GT 1 OR R902 GT 1 THEN R110=9 ELSE -
   R110=COMBINE R901(2),R902(2)
```

Método b. Use IFs, colocando um valor default de 9 no início.

```
R110=9
IF V5 EQ 1 AND V6 EQ 1 THEN R110=1
IF V5 EQ 1 AND V6 INLIST (2,3) THEN R110=2
IF V5 EQ 2 AND V6 EQ 1 THEN R110=3
IF V5 EQ 2 AND V6 INLIST (2,3) THEN R110=4
```

Método c. Use a função RECODE.

```
R110=RECODE V5,V6(1/1)=1,(1/2-3)=2,(2/1)=4,(2/2-3)=5,ELSE=9
```

10. Agregando casos com Recode. Suponha que nós desejamos analisar os dados (consistindo de registros a nível individual) no nível de aldeia, por exemplo, para produzir uma tabela mostrando a distribuição de renda dos habitantes (V8,V9) e % de pessoas possuindo um carro (V31) na aldeia. Nós poderíamos fazer isso utilizando AGGREG para agregar os dados a nível de aldeia e então executando TABLES. Alternativamente, nós poderíamos usar declarações CARRY, EOF e REJECT da linguagem Recode e usar TABLES diretamente.

```
 1         CARRY (R901,R902,R903,R904)
 2         IF (R901 EQ 0) THEN R901=V1
 3         IF (R901 NE V1) THEN GO TO VIL
 4         IF EOF THEN GO TO VIL
 5         R902=R902+1
 6         R903=R903+V8+V9
 7         IF (V31 EQ 1) THEN R904=R904+1
 8         REJECT
 9   VIL   R101=(R904*100)/R902
10         R101=BRAC(R101,<25=1,<50=2,<75=3,<101=4)
11         R102=R903/R902
12         R102=BRAC(R102,<1000=1,<2000=2,<5000=3,ELSE=4)
13         R901=V1
14         R902=1
15         R903=V8+V9
16         IF (V31 EQ 1) THEN R904=1 ELSE R904=0
17         NAME R102'renda média', R101'% owning car'
```

R901 é uma variável de trabalho usada para guardar a ID da aldeia corrente; quando o primeiro caso é lido (R901=0), o valor da ID (V1) da aldeia é designado para R901; R902 a R904 são variáveis de trabalho para, respectivamente, o número de pessoas na aldeia, a renda total das pessoas na aldeia e o número de pessoas possuindo carros na aldeia.

Enquanto a ID da aldeia permanece a mesma, dados são acumulados nas variáveis R902 a R904 (cujos valores são "transportados" a medida que casos novos são lidos). O caso é então rejeitado (não passado à análise) e o próximo caso é lido. Quando uma mudança na ID da aldeia é encontrada, as instruções na label VIL são executadas: o conteúdo corrente das variáveis R902, R903 e R904 é usado para computar as variáveis requeridas (renda média agrupada e % agrupado de donos de carro) e essas variáveis são então passadas para a análise depois de, primeiramente, reajustar os valores das variáveis de trabalho para os valores do último caso lido (o primeiro caso para a próxima aldeia). Quando o fim do arquivo é alcançado, nós precisamos ter certeza de que os dados da última aldeia é utilizado. A declaração 4 alcança esse objetivo.

4.16 Restrições

1. Número máximo de variáveis-R é 200.
2. Número máximo de tabelas numeradas (BRAC, RECODE, TABLE) é 20.
3. Número máximo de caracteres em uma declaração Recode excluido traços é 1024.

4. Número máximo declarações de labels é aproximadamente 60.
5. Número máximo de constantes, incluindo aquelas em todas as tabelas, é aproximadamente 1500.
6. Número máximo de nomes que podem ser definidos em declarações NAME é 70.
7. Número máximo de valores de dados perdidos que podem ser definidos em declarações MDCODES é 100 e apenas 2 casas decimais são mantidas nas variáveis-R.
8. Número máximo de aninhamento de parênteses dentro de uma declaração (i.e. parênteses dentro de parênteses) é 20.
9. Número máximo de operadores aritméticos é aproximadamente 400.
10. Número máximo de variáveis com a declaração SELECT é 50.
11. Número máximo de declarações IF é aproximadamente 100.
12. Número máximo de aninhamento de funções (i.e. declarações de funções como argumento de funções) é 25.
13. Número máximo de declarações é aproximadamente 200.
14. Número máximo de labels em uma declaração BRANCH é 20.
15. Número máximo de variáveis CARRY é 100.
16. O "Número máximo de variáveis" dado na seção "Restrições" de cada write-up de programa de análise inclui variáveis-R e -V usadas na análise e variáveis-V usadas em Recode mas não utilizadas na análise. Portanto, se um programa possui um máximo de 40 variáveis e 40 variáveis de entradasão utilizadas na análise, não se pode utilizar nenhuma outra variável de entrada, além das 40, nas declarações de Recode. Variáveis-R definidas nas declarações de Recode mas não utilizadas na análise não precisam ser contadas dentro do "número máximo de variáveis".
17. Filtragem ocorre antes da recodificação, de modo que variáveis de resultado não podem ser referenciadas em filtros principais.

4.17 Nota

Recodificação univariada/bivariada pode ser conseguida usando o método TABLE, IF ou RECODE. Abaixo segue uma breve comparação desses métodos levando-se em consideração dois aspectos de execução.

Compleitude

- TABLE...produz recodificação completa. Um valor de resultado é produzido mesmo quando o valor de entradaestá fora da tabela (dado que o valor default de ELSE é 99).
- RECODE permite recodificação parcial. Se nenhum teste é verdadeiro, e nenhum valor para ELSE é especificado, não ocorrerá recodificação.

Tamanho da tabela

- Recodificação univariada e bivariada de maiores proporções e completa são produzidas mais eficientemente por TABLE e IF...
- Em uma recodificação univariada, de grandes dimensões, um para um, usar uma linha de uma tabela retangular, TABLE é melhor que IF...

Capítulo 5

Gerenciamento e Análise de Dados

5.1 Validação de Dados com IDAMS

5.1.1 Resumo

Antes de se começar a análise de dados com qualquer software, os dados normalmente necessitam ser validados. Tal validação tipicamente é composta de três estágios:

1. Checando se os dados estão completos, i.e. verificando que todos os casos esperados estão presentes no arquivo de dados e que os registros corretos existem para cada caso se houver registros múltiplos por caso.
2. Checando se variáveis numéricas possuem apenas valores numéricos e checando se os valores são válidos.
3. Checagem de consistência entre variáveis.

Como muitos outros softwares estatísticos, IDAMS requer que haja a mesma quantidade de dados para cada caso. Se os dados para um caso se estende por vários registros, então cada caso deve constar do mesmo conjunto de registros. Se certas variáveis não são aplicáveis em alguns casos, então valores "perdidos" devem ser designados. Capacidade de checagem de fusão de registros em IDAMS permite checar se cada caso de dados possui o conjunto correto de registros. Isso é executado pelo programa MERCHECK, o qual produz um arquivo de saída "retangular" onde registros extras/duplicados foram deletados e casos com registros perdidos ou foram descartados ou preenchidos com registros dummy.

Checagem de valores não-numéricos em variáveis numéricas e a conversão opcional de campos em branco em valores numéricos específicos do usuário é executada pelo programa BUILD. Checagem de outros códigos inválidos é executada pelo programa CHECK onde os códigos válidos são definidos em declarações de controle especiais ou obtido de registros-C no dicionário que descreve os dados.

Se dados são carregados usando-se a Interface do Usuário do WinIDAMS, caracteres não-numéricos (exceto campos vazios) em campos numéricos não são permitidos. Além do mais, há a possibilidade de checagem de dados durante a entrada de dados e de uma checagem completa de códigos inválidos no arquivo de dados completo. Registros-C no dicionário são utilizados para essa finalidade.

Checagem de consistência pode ser expressa na linguagem de recodificação do IDAMS e pode ser usada com o programa CONCHECK para listar casos com inconsistências.

Erros encontrados em quaisquer desses passos podem ser corrigidos diretamnete pela Interface do Usuário ou usando o programa CORRECT do IDAMS. Uma sequência típica de passos para detectar e corrigir erros nos dados é descrita detalhadamente abaixo.

5.1.2 Checando a Integridade dos Dados

Passo 1 Produz tabelas de resumo mostrando a distribuição de casos ao longo de unidades amostrais, áreas geográficas, etc. para checar em relação aos totais esperados. Esse é um survey amostral

particularmente útil. Por exemplo, suponha que um survey de domicílios é realizado. Uma amostra é obtida ao selecionar-se primeiro unidades amostrais (PSU), até 5 áreas dentro de cada PSU e entrevistando domicílios nessas áreas. A distribuição de domicílios por PSU e por área nos dados pode ser produzido com a preparação de um pequeno dicionário contendo apenas as duas variáveis: PSU e área. A tabela se pareceria com algo como isso:

			V2 ÁREA				
			01	02	03	04	05
V1	PSU	01	3	6	2		
		02	10	4	2	8	5
		03					
		.					
		.					

Essa tabela poderia ser comparada com o livro dos entrevistadores para checar se os dados de todos as entrevistas realizadas existem no arquivo.

Passos 2, 3 e 4 são necessários apenas quando os casos são compostos de mais de um registro.

Passo 2 Os registros de dados "brutos" originais são classificados em ordem de identificação de caso/de registro usando o programa SORMER.

Passo 3 O arquivo de dados brutos classificados é checado com MERCHECK para ver se ele possui o conjunto de registros correto para cada caso. O arquivo de saída contém apenas casos "bons", i.e. aqueles com registros corretos. Registros extras e registros duplicados são descartados. Casos com registros perdidos são, ou descartados, ou preenchidos com espaços em branco. Todos os casos com erro de fusão são listados.

Passo 4 Correções são agora feitas nos erros detectados por MERCHECK. Isso pode ser feito de diversas maneiras:

- Re-entrar com casos "ruins" e fundi-los com o arquivo de saída de MERCHECK usando SORMER.
- Corrigir o dados brutos originais com um editor e refazer os passos 2 e 3.
- Re-entrar os casos "ruins", executar passos 2 e 3 e então fundir o resultado dessa execução do passo 3 com o resultado original do passo 3.

Qualquer que seja o método selecionado, MERCHECK deve ser re-executado no arquivo corrigido para ter-se certeza que todos os erros foram tratados.

5.1.3 Checando Valores Não-numéricas e Inválidas

Passo 5 Prepare um dicionário para todas as variáveis com as instruções apropriadas de lidar com campos em branco. Execute BUILD. Um dataset IDAMS é produzido (arquivos de Dados e Dicionário). Todos os valores não-numéricos inesperados são convertidos em 9's e reportados nos resultados.

Passo 6 Usando TABLES, imprima freqüências de distribuição de todas as variáveis qualitativas e valores de mínimo, de máximo e de média para variáveis quantitativas. Isso dá uma idéia inicial do conteúdo dos dados e mostra que variáveis possuem códigos inválidos (variáveis qualitativas) ou valores muito grandes/pequenos (variáveis quantitativas). Isso pode também ser comparado mais tarde com uma distribuição similar e valores obtidos após a limpeza para ver como a validação de dados afetou os dados.

Passo 7 Prepare declarações de controle especificando os códigos válidos ou intervalo de valores para cada variável. Esses podem ser preparados antes do tempo para todas as variáveis ou alternativamente, depois do passo 6 para apenas aquelas variáveis que sabe-se conter códigos inválidos. Use o dataset produzido no passo 5 como entrada para o programa CHECK para obter uma lista de casos com valores inválidos. Note que a especificação de códigos válidos para variáveis pode também ser obtida dos registros-C no dicionário se estes forem introduzidos no passo 5.

Passo 8 Prepare correções para erros detectados no passo 5 e passo 7. Use o programa CORRECT para atualizar o dataset IDAMS criado no passo 5.
Note que as correções podem também ser feitas com a Interface do Usuário do WinIDAMS se o número de casos não for tão grande. Contudo, usando CORRECT é um método menos propenso a erros.

Execute passos 7 e 8 até que nenhum erro seja reportado.

5.1.4 Checagem de Consistência

Passo 9 Prepare declarações lógicas das checagens de consistência a serem executadas, e.g.
GRÁVIDA (V32) = inaplicável se e somente se SEXO (V6) = Masculino.
Designe um número de "resultado" para cada checagem de consistência e traduza a lógica em declarações de Recode onde o resultado é igualado a 1 para cada inconsistência, e.g.

```
IF V6 EQ 1 AND V32 NE 9 THEN R1001=1
IF V6 NE 1 AND V32 EQ 9 THEN R1001=1 ELSE R1001=0
```

Use o conjunto de declarações de Recode com CONCHECK para imprimir casos com erros.

Passo 10 Corrija casos com erros como no passo 8.

Executa os passos 9 e 10 até que nenhum erro seja reportado. O resultado dos dados da execução final do CORRECT estará pronta para análise.

5.2 Gerenciamento e Transformação de Dados

IDAMS contém um conjunto extenso de facilidades para gerar índices, medidas derivadas, agregações, e outras transformações dos dados, incluindo recodificação alfabética. As capacidades mais freqüêntemente utilizadas são propiciadas pela facilidade Recode, que pode executar operações temporárias em todos os programas de análise que utilizam dataset IDAMS. Resultados de recodificação podem ser salvos como variáveis permanentes através do programa TRANS. Essas facilidades operam em variáveis dentro de um caso e permitem recodificação dos valores de uma ou mais variáveis, geração de variáveis via combinação de variáveis, controle da seqüência dessas operações através de testes de expressões lógicas, e um número de declarações especiais e funções. A informação do novo dicionário necessária para descrever os resultados das operações executadas é automaticamente produzida.

Para agregação dos casos, o programa AGGREG está disponível. AGGREG propicia somas aritméticas e medidas relacionadas, intervalos, e contagens de valores válidos de dados ao longo de grupos de casos. Uso típico de AGGREG envolve o uso anterior do programa SORMER para ordenar o arquivo Dados em grupos desejados.

Há um número de circustâncias nas quais é necessário combinar os registros de dois arquivos diferentes, por exemplo, dados coletados em diferentes pontos no tempo. A medida que os valores das variáveis para cada nova onda são recebidos, o objetivo é adicioná-los ao registro contendo todos os dados anteriores para o mesmo respondente ou caso. O programa MERGE alcançará isso, incluindo-se espaços em branco apropiados onde os respondentes não forem encontrados nessa nova onda de coleta de dados. Exemplos similares ocorrem quando resíduos ou outra forma de scores de escala são gerados para cada caso por um programa de análise e precisam ser incluído nos dados originais.

Um processo de combinação ligeiramente diferente ocorre quando dados de diferentes níveis de análise devem ser combinados. Uma ilustração disso é a adição de dados de domicílio aos registros de respondentes individuais. Quando um dataset é ordenado de maneira que os respondentes do mesmo domicílio estão juntos, MERGE propiciará a necessária fusão dos registros duplicados. Um situação similar ocorre quando resumos de grupos de AGGREG devem ser adicionados aos registros de cada caso em cada grupo respectivo.

Outro processo de combinação de datasets, às vezes conhecido por fusão, ocorre quando casos adicionais devem ser incluídos em um dataset. Os novos registros devem ser descritos pelo mesmo dicionário dos dados originais. Esse tipo de fusão pode ser alcançado com o programa SORMER.

Funções para estabelecer subconjuntos estão disponíveis como operações temporárias em muitos programas do IDAMS (utilizando-se um "filtro") para selecionar casos particulares para processamento. Arquivos permanentes contendo subconjuntos de datasets IDAMS (um subconjunto de variáveis ou subconjunto de casos, ou ambos) podem também ser criados. Os programas SUBSET e TRANS serão muito provavelmente utilizados nesses casos, apesar de que vários outros programas que produzem datasets, como MERGE, podem também ser usados. Seleção de casos pode ser feita assumindo-se que apenas certos casos são logicamente de interesse (como o caso de respondentes do sexo feminino apenas), ou de modo aleatório, usando a função RAND do Recode com o programa TRANS.

Uma mostra do valores reais armazenados em um dataset IDAMS é, muitas vezes, de grande ajuda para checagem de resultados dos passos de modificação dos dados e, de fato, em qualquer estágio. O programa LIST está disponível para esse propósito, e permite listagems completas de uma seleção de casos e variáveis específicas. A seleção ou filtragem de casos para exposição pode ser feito usando combinações de várias variáveis em expressões lógicas; um exemplo seria a seleção apenas de registros de mulheres solteiras entre 21 e 25 anos de idade. Variáveis numéricas e alfabéticas de um dataset, como também variáveis construídas com declarações de Recode podem ser listadas. A Interface do Usuário possui também uma opção para imprimir dados em formato de tabela.

5.3 Análise de Dados

A consideração fundamental para o usuário ao selecionar programas de análise é se as funções estatísticas apropriadas são fornecidas. Um guia em relação a este assunto está além do escopo desse manual. Um resumo das funções de cada programa de análise do IDAMS pode ser encontrado na Introdução. Mais detalhes são dados nos write-ups dos programas individuais. As fórmulas utilizadas para computar as estatísticas em cada programa e referências são dadas nos capítulos relevantes na parte "Fórmulas Estatísticas e Referências Bibliográficas".

5.4 Exemplo de uma Pequena Tarefa a ser Executada pelo IDAMS

Suponha que um dataset IDAMS contém respostas de um questionário de survey e inclui as seguintes variáveis:

V11 dá o sexo do respondente de acordo com o seguinte código:

1. Masculino 2. Feminino 9. Não revelado

V12 é a renda do respondente em dólares (99999 = não revelado).

V13 até V16 são medidas atitudinais em diferentes assuntos. Cada variável é codificada para refletir os sentimentos dos respondentes da seguinte maneira:

1. Muito positivo 2. Positivo 3. Neutro 4. Negativo 5. Muito negativo 8. Não sabe
9. Não revelado 0. A questão é irrelevante para esse respondente

Suponha que apenas um agrupamento ou recodificação de níveis de renda da seguinte maneira é necessário:

Novo código	Significado
1	Renda no intervalo \$0 a \$9999
2	Renda no intervalo \$10,000 a \$29,999
3	renda \$30,000 ou mais
9	Recusado, Não revelado, Não sabe

Desejam-se tabulações-cruzadas entre a versão recodificada da variável renda, V12, e cada uma das variáveis atitudinais, V13 a V16. Apenas respondentes do sexo feminino devem ser selecionados para essa análise.

Um "setup" do IDAMS contendo todas as declarações de controle necessárias para esse trabalho é mostrado abaixo. Os números em parêntese na esquerda identificam cada declaração de controle e ligam as declarações à explicação subseqüênte.

```
(1)  $RUN TABLES
(2)  $FILES
(3)  DICTIN = ECON.DIC
(4)  DATAIN = ECON.DAT
(5)  $RECODE
(6)      R101=BRAC(V12,0-9999=1,10000-29999=2,30000-99998=3, -
(7)           ELSE=9)
(8)      NAME R101 'RENDA AGRUPADA'
(9)  $SETUP
(10) INCLUDE V11=2
(11) EXEMPLO DE TABELA USADO DADOS ECONOMICOS
(12) *
(13) TABLES
(14) ROWVARS=(R101,V13-V16)
(15) ROWVAR=R101  COLVARS=(V13-V16) CELLS=(FREQS,ROWPCT)  STATS=CHI
```

Resumidamente, isso é o que cada declaração faz:

(1) "$RUN TABLES" é um comando de IDAMS especificando que o programa TABLES deve ser executado.

(2) Essa declaração sinaliza o início das definições de arquivo para execução.

(3)&(4) O dataset IDAMS é armazenado em dois arquivos separados. Um contém o dicionário, o outro os dados.

(5) Essa declaração sinaliza que transformações dos dados são requeridas. As declarações seguintes a isso são comandos específicos da facilidade Recode.

(6)(7) Essas duas linhas (uma original e uma continuação) formam uma declaração da facilildade Recode indicando o agrupamento desejado para variávelde renda, V12, seguindo o esquema explicitado antes. O resultado da função BRAC é armazenado na variável R101.

(8) Essa declaração designa um nome para a variável R101.

(9) "$SETUP" é um comando que indica o fim das declarações de Recode e que declarações de controle do programa TABLES seguem.

(10) Isso é um "filtro" que declara que os únicos casos de dados que devem ser usados são aqueles onde a variável V11 tem um valor de código de 2, para feminino.

(11) Isso é um título que contém o texto a ser usado nos resultados.

(12) Essa linha especifica os pincipais parâmetros. Como apenas o asterisco é dado, todas as opções default para os parâmetros são escolhidas para a execução corrente.

(13) A palavra TABLES é posta aqui para separar a informação global precedente para a execução inteira das especificações para tabelas individuais que vêm a seguir.

(14) Essa declaração requer distribuições de freqüência univariadas para 5 variáveis.

(15) Agora tabelas bivariadas (2-way) são requisitadas. As células devem conter as contagens (freqüências) e percentagens das linhas; uma estatística chi-quadrado será impressa para cada tabela. As 2 listas de variáveis seguintes às palavras-chave ROWVAR e COLVARS especificam as variáveis que serão utilizadas para as linhas e colunas das tabelas, respectivamente. Quatro tabelas serão produzidas: R101 (renda agrupada) por V13, V14, V15 e V16).

Parte II

Trabalhando com o WinIDAMS

Capítulo 6

Instalação

6.1 Requerimentos do Sistema

- O software WinIDAMS está disponível para as versões 32-bit dos sistemas operacionais Windows (Windows 95, 98, NT 4.0, 2000 e XP).
- Um pentium II ou processador mais rápido e 64 megabytes RAM são recomendados.
- Em todos os sistemas, você deve possuir 11 megabytes de espaço de disco livre antes de tentar instalar o software WinIDAMS em cada linguagem.

6.2 Procedimento de Instalação

- WinIDAMS 1.3 é armazenado em CD em um arquivo auto-executável

```
WinIDAMS\English\Install\WIDAMSR13E.EXE      : Versão em Inglês
WinIDAMS\French\Install\WIDAMSR13F.EXE       : Versão em Francês
WinIDAMS\Spanish\Install\WIDAMSR13S.EXE      : Versão em Espanhol
WinIDAMS\Portuguese\Install\WIDAMSR13P.EXE   : Versão em Português
```

  ou em um arquivo de download equivalente.

- Para instalar a versão em Português:
  1. Selectione `WIDAMSR13P.EXE` com o Windows explorer.
  2. Dê dois clicks nesse arquivo e siga as mensagens.
  3. No final do procedimento de instalação, uma caixa de diálogo aparecerá perguntando: "Você deseja instalar a atualização do HTML Help 1.3 agora?". É recomendado responder SIM.
- O procedimento de instalação cria dois itens no menu Program Manager/Start, um para executar WinIDAMS e um para desinstalar WinIDAMS. Ele também cria um ícone no desktop que é um link/atalho para WinIDAMS.

6.3 Testando a Instalação

Um arquivo Setup contendo instruções para execução de 4 programas de gerenciamento de dados (CHECK, CONCHECK, TRANS e AGGREG) e 6 programas de análise de dados (TABLES, REGRESSN, MCA, SEARCH, TYPOL e RANK) são copiados dentro do folder Trabalho ("work") durante a instalação. Para executar isso:

- Inicie WinIDAMS com um click duplo no seu ícone.
- Você verá a janela principal do WinIDAMS com uma aplicação default exposta no painel esquerdo. Abra o folder Setups. Há um arquivo demo.set com instruções para execução dos 10 programas.
- Com o click duplo, o arquivo se abre na janela Setup. Execute-o dessa janela. Resultados da execução são enviados para o arquivo idams.lst que é imediatamente aberto na janela Resultados.
- A versão distribuída dos resultados é encontrada no arquivo demo.lst no folder "Results".
- Compare as duas versões dos resultados.

6.4 Folders e Arquivos Criados Durante Instalação

6.4.1 Folders do WinIDAMS

O nome completo do atalho do folder do Sistema WinIDAMS é dado no "Selecione Diretório de Destino" do programa de instalação e os seguintes folderes são criados durante a instalação (ver o capítulo "Arquivos e Folders" para detalhes):

Versão em Inglês	Versão em Francês
<WinIDAMS13-EN>\appl	<WinIDAMS13-FR>\appl
<WinIDAMS13-EN>\data	<WinIDAMS13-FR>\data
<WinIDAMS13-EN>\temp	<WinIDAMS13-FR>\temp
<WinIDAMS13-EN>\trans	<WinIDAMS13-FR>\trans
<WinIDAMS13-EN>\work	<WinIDAMS13-FR>\work

Versão em Espanhol	Versão em Português
<WinIDAMS13-SP>\appl	<WinIDAMS13-PT>\appl
<WinIDAMS13-SP>\data	<WinIDAMS13-PT>\data
<WinIDAMS13-SP>\temp	<WinIDAMS13-PT>\temp
<WinIDAMS13-SP>\trans	<WinIDAMS13-PT>\trans
<WinIDAMS13-SP>\work	<WinIDAMS13-PT>\work

6.4.2 Arquivos Instalados

Arquivos de sistema no folder Sistema

(\WinIDAMS13-EN, \WinIDAMS13-FR, \WinIDAMS13-SP, \WinIDAMS13-PT)

WinIDAMS.exe	Arquivo executável principal para a Interface do Usuário do WinIDAMS
Ter32.dll	|
Hts32.dll	| Dlls usada pela Interface do Usuário do WinIDAMS
unesys.exe	Arquivo executável usado para processamento de setups
Idame.mst	Arquivo Master da base de dados de texto para programas do IDAMS
Idame.xrf	Arquivo de referência cruzada da base de dados de texto para programas do IDAMS
idams.def	Definição do mapeamento entre ddnames e nomes de arquivos
Graph32.exe	Arquivo executável do GraphID
graphid.ini	Arquivo Ini utilizado por GraphID para armazenar cores, fontes e coordenadas
Idtml32.exe	Arquivo executável do TimeSID
idaddto32.dll	Dll usada por GraphID e TimeSID
IDAMSC_DLL.dll	Dll usada por TimeSID
Idams.chm	Arquivo de ajuda do manual do WinIDAMS
<pgmname>.pro	Protótipos para progamas do IDAMS

Arquivos de dicionário e de dados utilizados para os exemplos no folder Dados ("data")

(\WinIDAMS13-EN\data, \WinIDAMS13-FR\data, \WinIDAMS13-SP\data, \WinIDAMS13-PT\data)

```
educ.dic
educ.dat
rucm.dic
rucm.dat
watertim.dic
watertim.dat
data.csv
tab.mat
```

Setup de demonstração e arquivo de resultados no folder Trabalho ("work")

(\WinIDAMS13-EN\work, \WinIDAMS13-FR\work, \WinIDAMS13-SP\work, \WinIDAMS13-PT\work)

```
demo.set
demo.lst
```

6.5 Desinstalação

Um programa desinstalador é criado durante o procedimento de instalação. O usuário pode executar o desinstalador ao clicar no WinIDAMS13-PT/Uninstall WinIDAMS13-PT no menu Program Manager/Start ou ao deletar a entrada "WinIDAMS Versão 1.3 em Português, Janeiro de 2005" no applet Add/Remove Programs Control Panel. O desinstalador deleta o conteúdo do folder do WinIDAMS selecionado durante o processo de instalação. Ele não deleta folders que não estejam vazios.

Capítulo 7

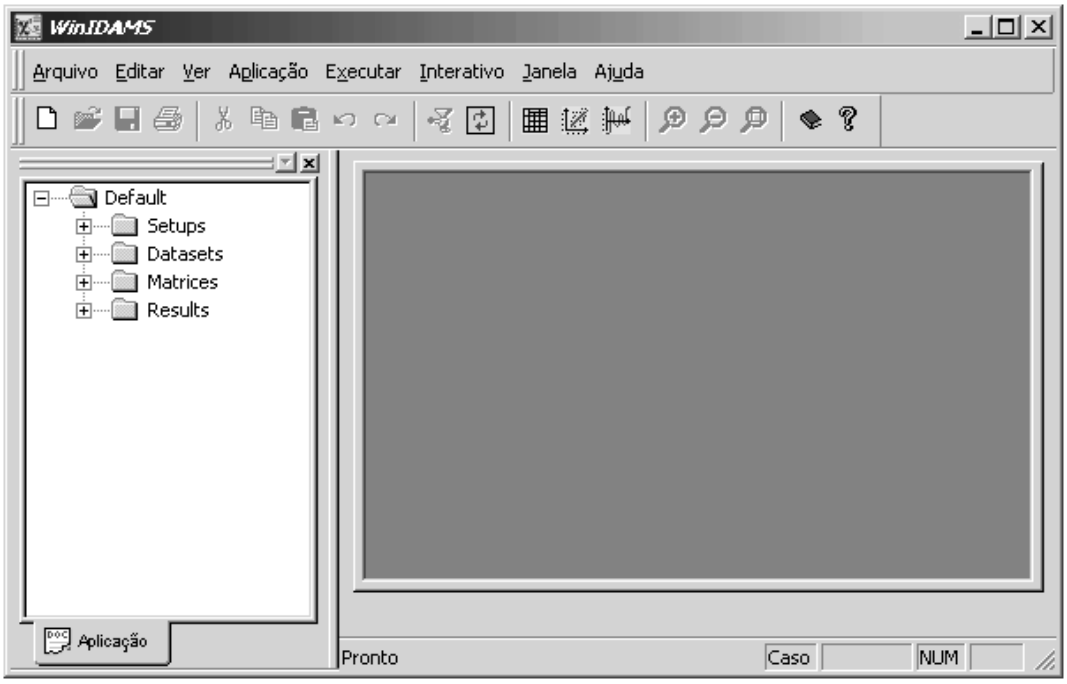
Iniciando

7.1 Visão Geral dos Passos a serem Executados com o WinIDAMS

Nesse exemplo, um dicionário do IDAMS para a descrição de dados coletados por um questionário é preparado e os dados de alguns respondentes são carregados. Um conjunto de declarações de controle do IDAMS (um "setup") é então preparado e utilizado para produzir distribuições de freqüência para Idade, Sexo e Educação (número de anos) dividida em 4 grupos. Os passos necessários são os seguintes:

1. Criar um ambiente de aplicação.
2. Preparar e armazenar um dicionário descrevendo as variáveis nos dados.
3. Entrar os dados (esse passo seria eliminado se os dados tivessem sido preparados fora do WinIDAMS).
4. Preparar e armazenar um "setup" de instruções especificando o que deve ser feito com os dados.
5. Executar o programa do IDAMS como especificado no setup.
6. Rever os resultados e modificar o setup se necessário; então repetir a partir do passo 4
7. Imprimir os resultados.

Para começar, primeiro inicie o WinIDAMS. Você verá a janela Principal do WinIDAMS.

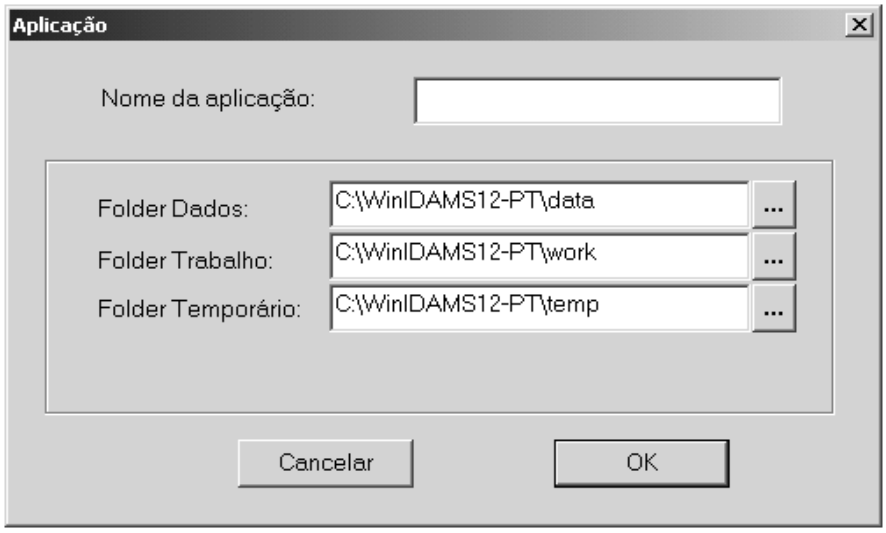


7.2 Criar um Ambiente de Aplicação

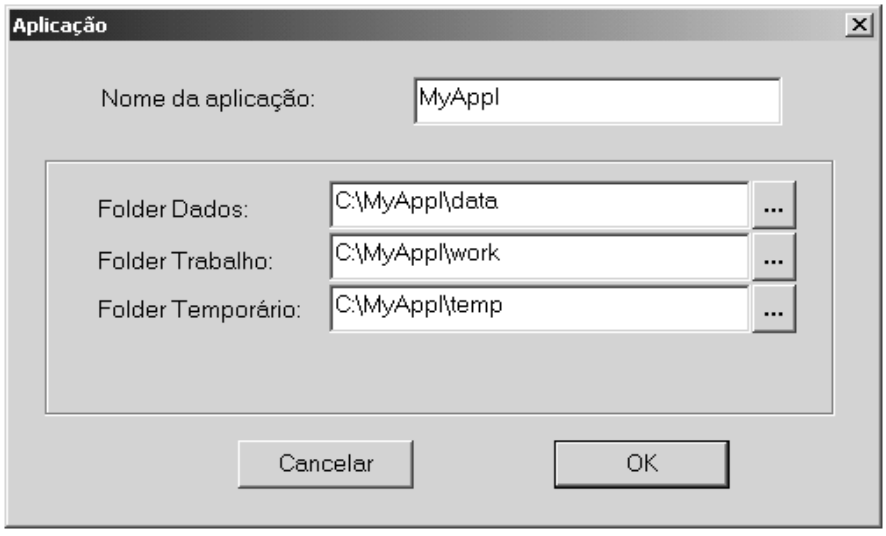
O ambiente de aplicação permite que você pré-defina atalhos completos para três folders. Todos os arquivos de entrada/saída serão abertos/criados por default em um desses folders. Isso evita que você entre com o nome completo do atalho do folder.

- Os arquivos Dados e Dicionário: no folder Dados.
- Os arquivos Setup e Resultados: no folder Trabalho.
- Os arquivos temporários: no folder Temporário.

Click em Aplicação na barra de menu e então em Novo. Agora você vê o seguinte diálogo:



Nós vamos criar uma nova aplicação com o nome "MyAppl" e com os folders de aplicação `C:\MyAppl\data`, `C:\MyAppl\work` e `C:\MyAppl\temp` entrando esses nomes nas caixas de texto correspondentes.



Para cada folder de aplicação introduzido que não existia, você verá um diálogo como este:



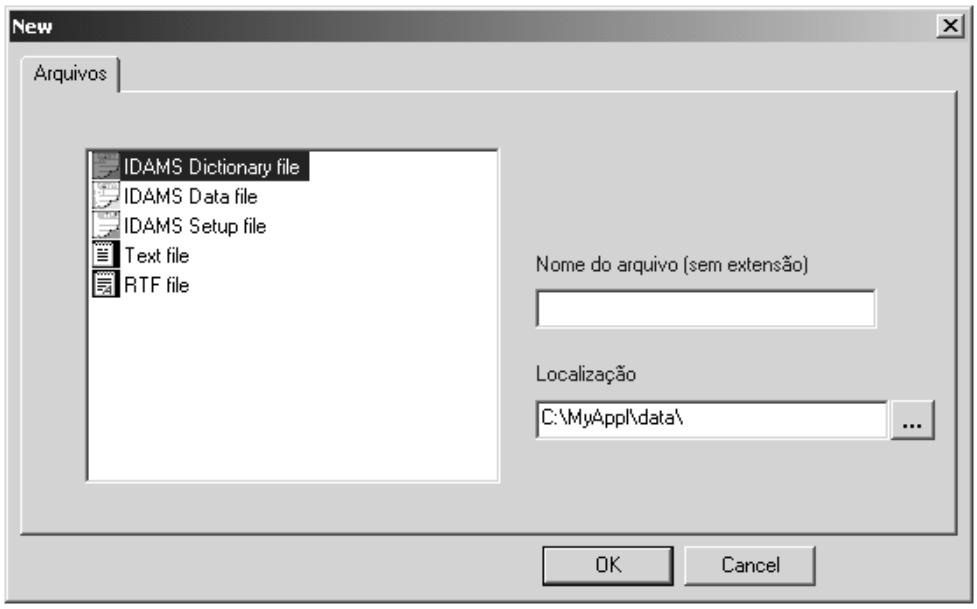
Clique em Yes para cada novo folder e então clique OK. Agora você verá a janela Principal do WinIDAMS novamente.

7.3 Preparar um Dicionário

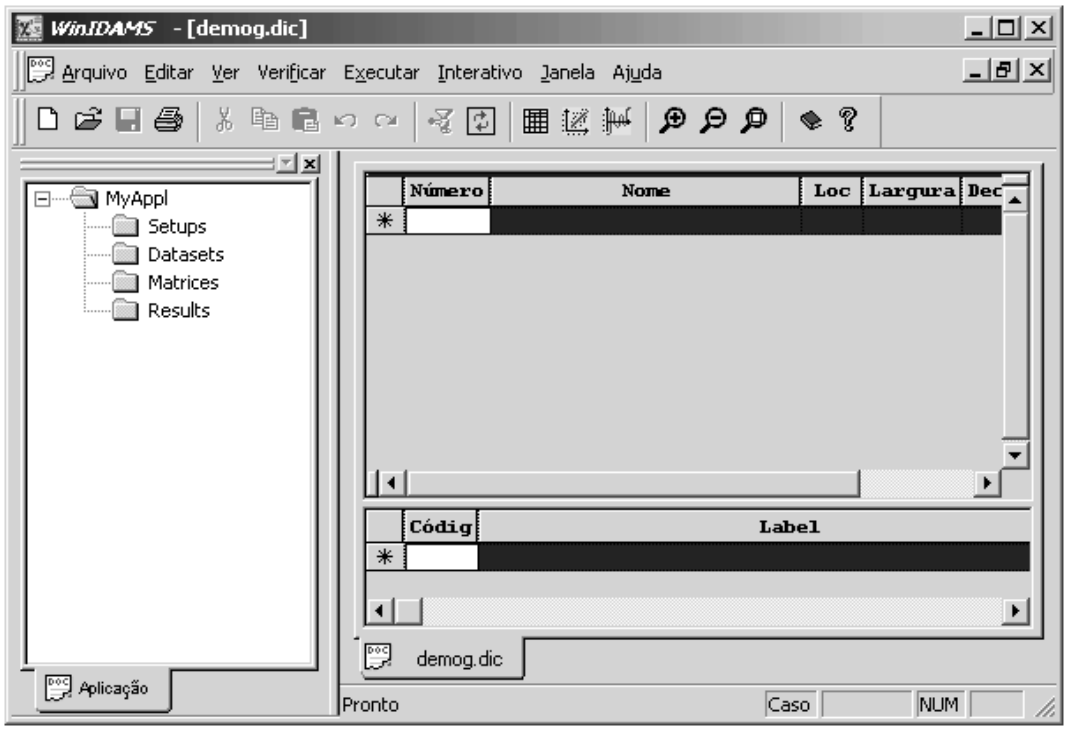
Nós criaremos um dicionário para descrever registros de dados contendo as seguintes variáveis:

Número	Nome	Largura	Código de Dados Perdidos
1	Identificação	3	
2	Idade	2	
3	Sexo	1	9
	1 Masculino		
	2 Feminino		
	9 DP		
4	Educação	2	

- Pressione Ctrl/N ou clique em Arquivo/Novo. Esses comandos abrem o diálogo Novo:



- O diálogo mostra a lista de tipos de documentos usados no WinIDAMS. Escolha “IDAMS Dictionary file”, já selecionado por default.
- Clique no campo de Nome de archivo e entre com o nome “demog”. Então clique OK. Note que a extensão .dic é adicionada automaticamente ao nome do arquivo.
- Você verá agora:
  - a janela Aplicação;
  - uma janela com 2 espaços para entrar as descrições das variáveis e labels e códigos associados opcionais. O nome do arquivo Dicionário completo “demog.dic” é mostrado na tela.



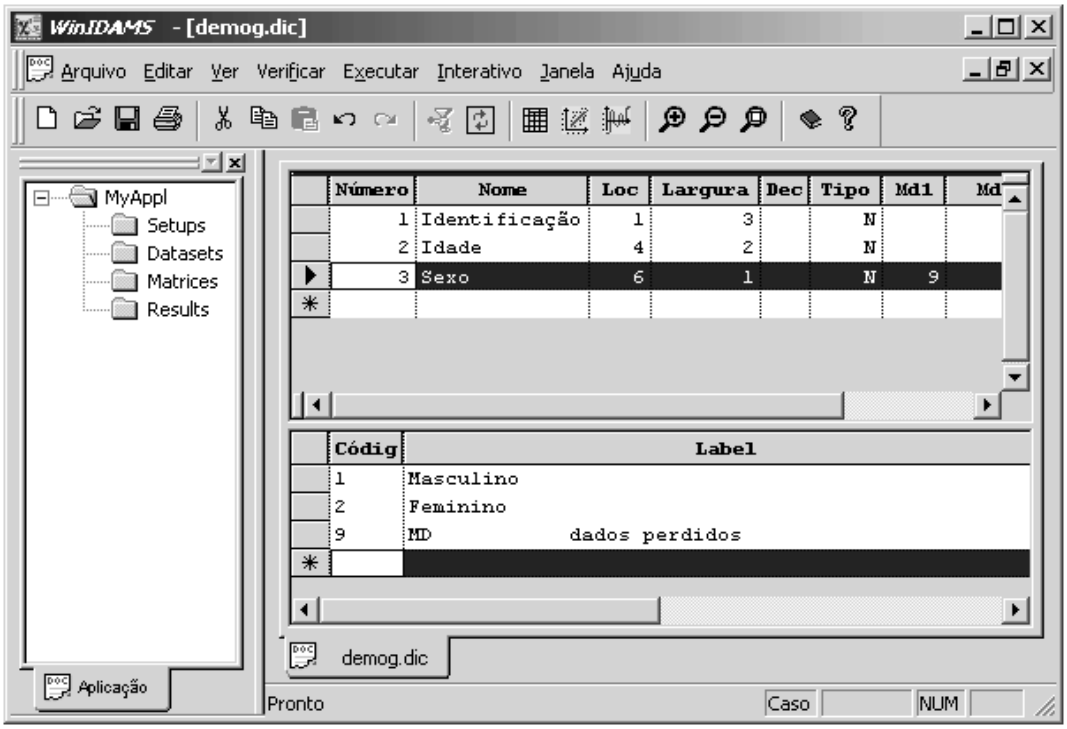
- Clique na primeira célula na linha do espaço de descrição de variáveis e entre o número da primeira variável. Tão logo você comece a entrar com informação na linha marcada com um asterisco, uma nova linha será criada logo após a linha corrente e a linha que você está editando mostrará um lápis no cabeçalho da linha. Ao pressionar Enter ou Tab você se moverá para o próximo campo. Agora entre o nome da variável e a largura. Pule o resto dos outros campos pressionando Enter ou Tab e aceite as descrições pressionando Enter ou Tab no último campo. Note que a localização default é dada pelo WinIDAMS quando a linha de descrição da variável for aceita.

- Quando você pressiona Enter ou Tab no último campo, o lápis desaparece, o que significa que a linha foi aceita depois de uma checagem rudimentar dos campos. O campo corrente é agora o primeiro campo da próxima linha (marcada com um asterisco) e você pode entrar com a descrição da 2ª variável, Idade. Faça o mesmo para variável 3, Sexo, mas dê a essa variável um código de MD1 (dados perdidos) de 9 (o código de não-resposta).

- Depois de aceitar a descrição da variável 3, o primeiro campo (número da variável) da linha com um asterisco se torna o campo corrente. Clique em qualquer campo da linha que se acabou de entrar (variável 3, Sexo) para fazê-la a linha corrente.

- Troque para a janela de códigos e suas labels clicando no campo de código na primeira linha. Note que essa janela é sincronizada com as variáveis selecionadas na janela de descrição de variáveis.

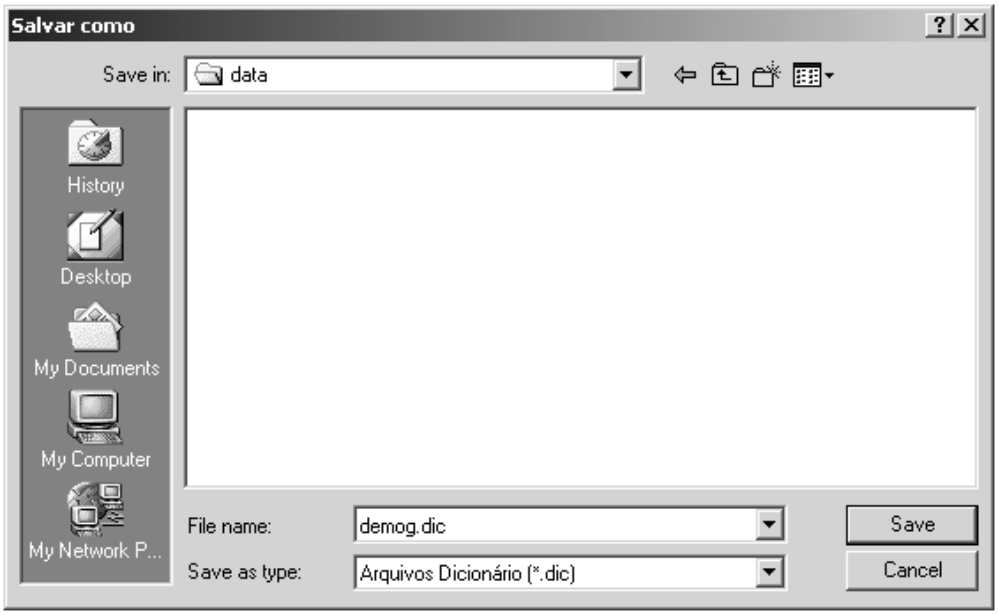
- Entre 1 no campo de código. Novamente, assim que você começar a entrar a label do código, uma nova linha com um asterisco é criada logo após a linha corrente e a linha que você está editando mostrará um lápis. Pressione Enter para se mover para o próximo campo, entre Masculino no campo label. Pressione Enter. O campo corrente é agora o campo de código da próxima linha e você pode entrar código 2 com a label Feminino e, similarmente, código 9.



- Volte para a janela de descrição de variáveis clicando no campo de número da variável da linha com um asterisco. Entre a informação da variável 4.

  Para deletar linhas, clique ao lado da linha e selecione Cortar no menu Editar.

- Salve o dicionário clicando em Arquivo/Salvar como, e aceite o nome do arquivo Dicionário "demog.dic".



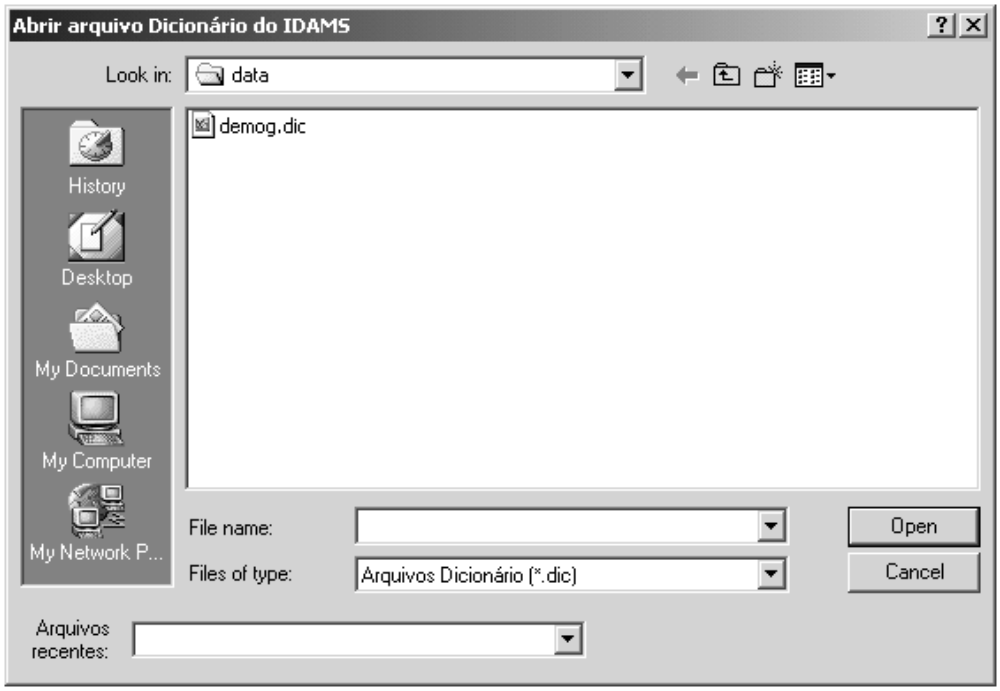
7.4 Entre com Dados

- Pressione Ctrl/N ou clique em Arquive/Novo. O mesmo diálogo Novo documento que foi visto acima para o dicionário é mostrado.

- Selecione o item "IDAMS Data file" da lista e entre com o nome do arquivo Dados. Por convenção, é melhor usar o mesmo nome para o arquivo Dados e o arquivo Dicionário que descreve os dados.

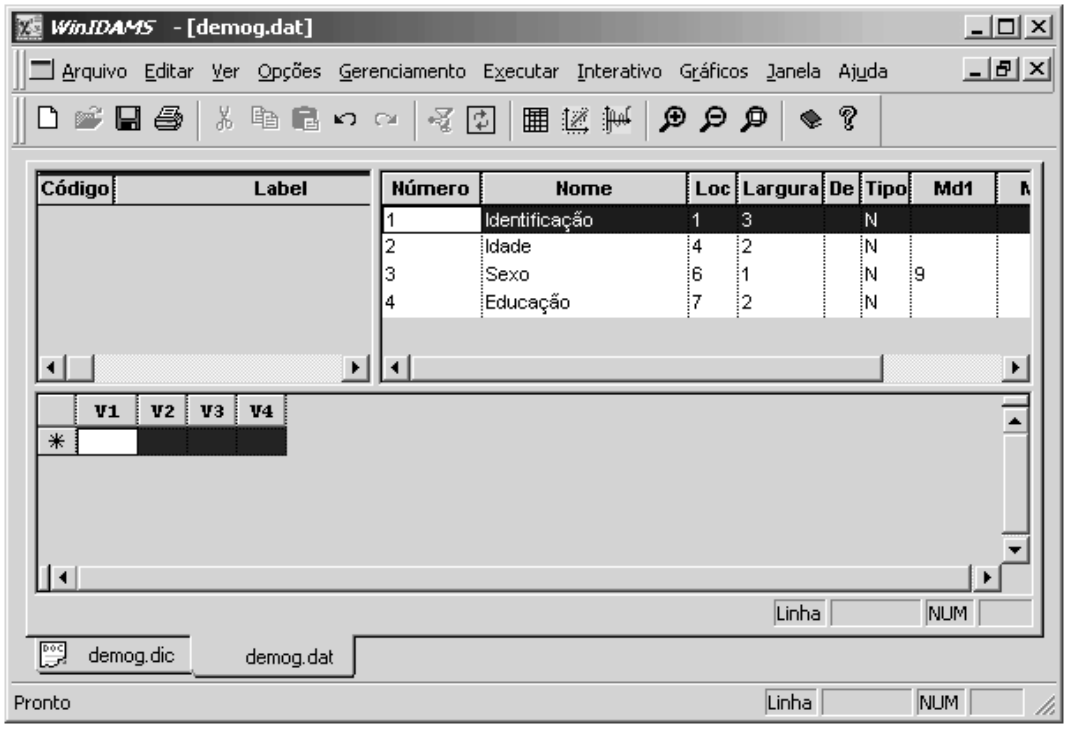
Somente a extensão do arquivo muda, ".dic" para o arquivo Dicionário e ".dat" para o arquivo Dados. O dicionário e dados constituem um dataset IDAMS. Entre "demog" como nome de arquivo e clique em OK.

- Um diálogo Arquivo Abrir exibe agora os dicionários que existem para a aplicação ativa e pede que você selecione o dicionário que descreve os dados. Selecione "demog.dic" e clique Abrir.

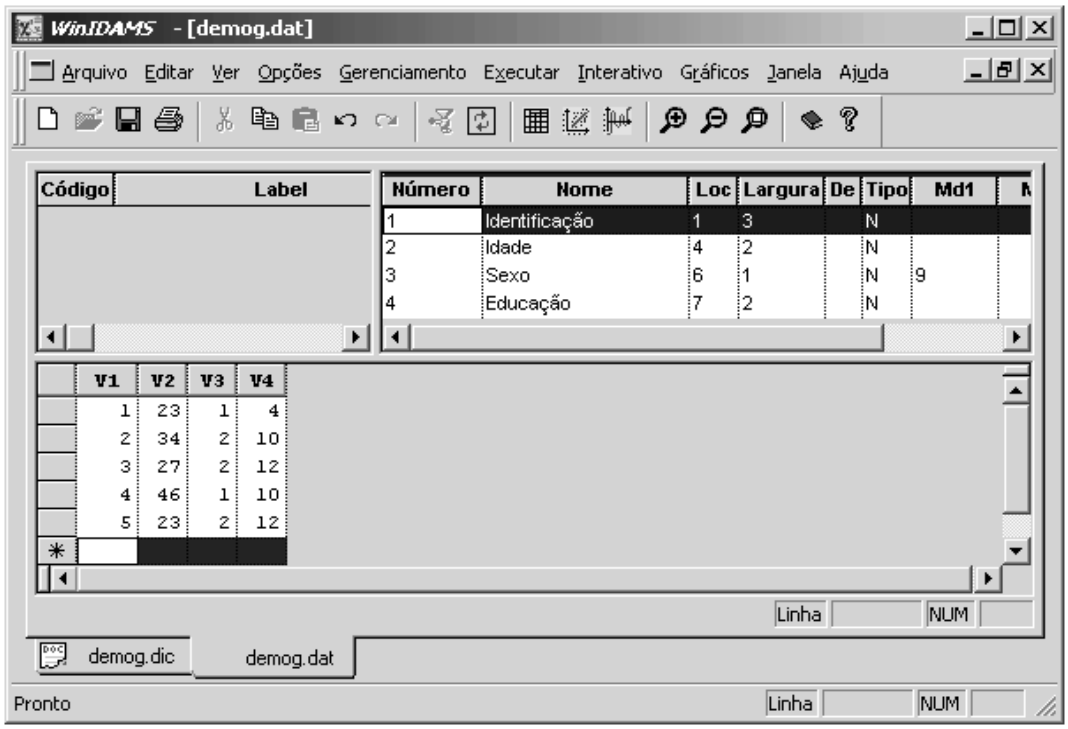


- Uma janela com três espaços agora aparece. Você deve entrar com os dados somente no espaço mais em baixo. Os outros dois espaços estão sincronizados para exibir a descrição da variável corrente e as labels do código, se existir. O nome completo do arquivo Dados "demog.dat" (a extensão .dat é adicionada automaticamente) é exibida na etiqueta.

  Note que nas ilustrações apresentadas abaixo a janela Aplicação foi fechada.


WinIDAMS - [demog.dat]
Arquivo Editar Ver Opções Gerenciamento Executar Interativo Gráficos Janela Ajuda
Código Label
Número Nome Loc Largura De Tipo Md1
1 Identificação 1 3 N
2 Idade 4 2 N
3 Sexo 6 1 N 9
4 Educação 7 2 N
V1 V2 V3 V4
Linha NUM
demog.dic demog.dat
Pronto
Linha NUM

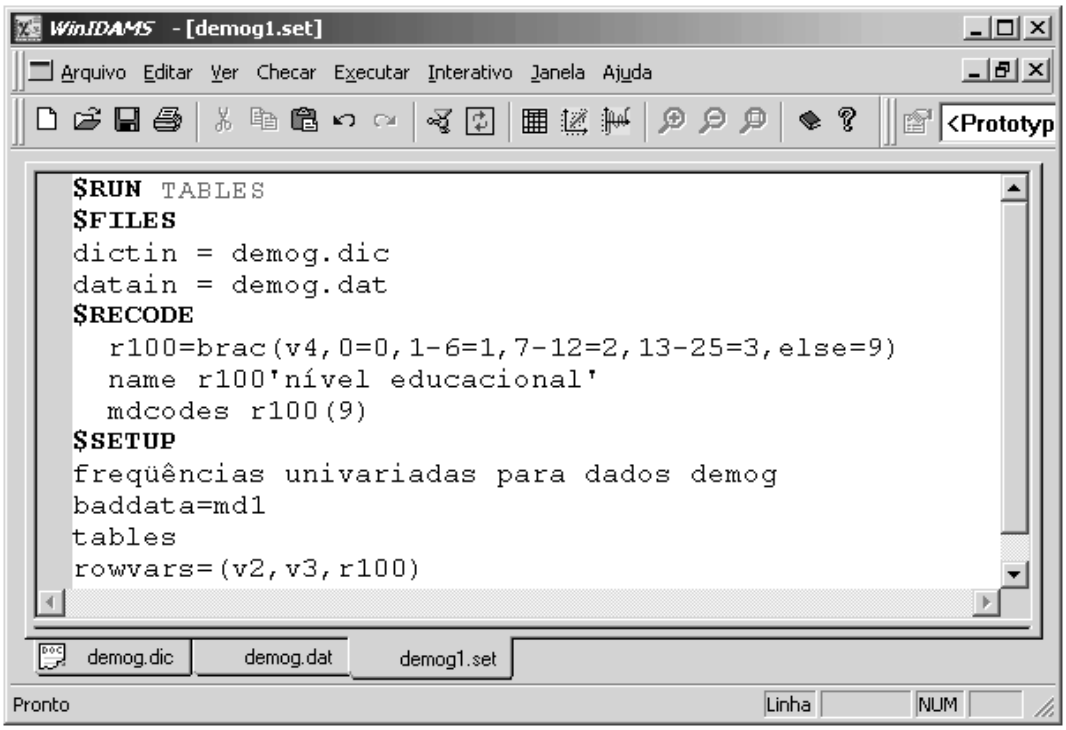
- Clique no primeiro campo da linha com um asterisco e digite a primeira linha de dados como mostrado abaixo, pressionando a tecla Enter depois de entrar cada valor de dados. Tão logo você comece a entrar com dados, uma nova linha é criada logo depois da linha corrente e o cabeçalho da linha corrente exibe um lápis, o que significa que você está editando essa linha.
- Depois de entrar com o valor da última variável V4 e pressionar Enter, o primeiro campo da próxima linha se torna o campo corrente.
- Entre os dados para os 5 casos abaixo.


WinIDAMS - [demog.dat]
Arquivo Editar Ver Opções Gerenciamento Executar Interativo Gráficos Janela Ajuda
Código Label
Número Nome Loc Largura De Tipo Md1
1 Identificação 1 3 N
2 Idade 4 2 N
3 Sexo 6 1 N 9
4 Educação 7 2 N
V1 V2 V3 V4
1 23 1 4
2 34 2 10
3 27 2 12
4 46 1 10
5 23 2 12
Linha NUM
demog.dic demog.dat
Pronto
Linha NUM

- Clique em Arquivo/Salvar para salvar os dados no arquivo “demog.dat”.

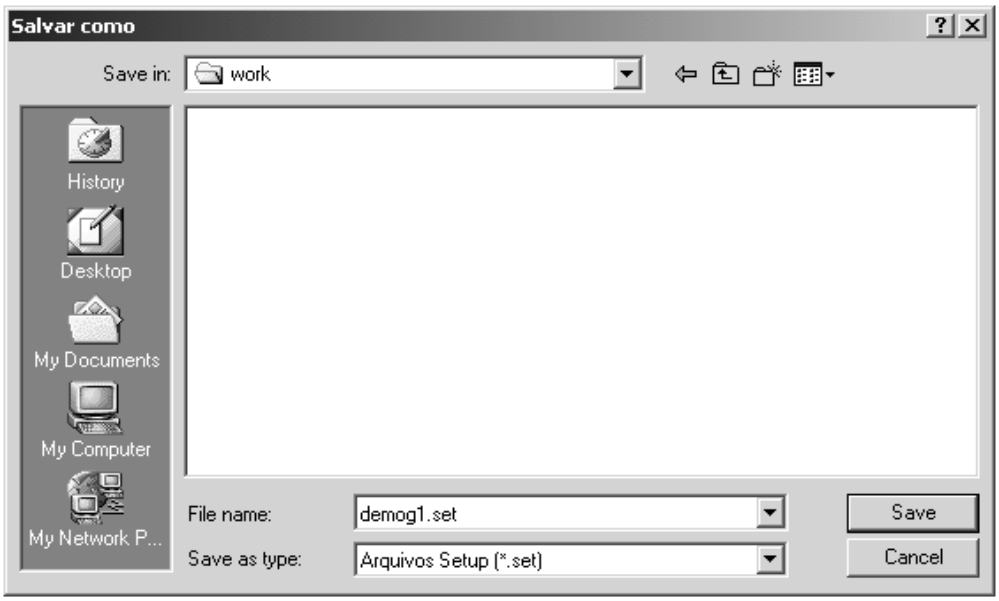
7.5 Prepare o Setup

- Pressione Ctrl/N ou clique em Arquivo/Novo.
- Selecione o item “IDAMS Setup file” da lista e entre com um nome, e.g. “demog1” para o arquivo de Setup. Clique OK. Note que a extensão .set é adicionada automaticamente ao nome de arquivo e o nome de arquivo completo “demog1.set” é exibido na etiqueta.
- Você agora verá uma janela vazia para entrar com o setup. Digite o seguinte:



O $RUN identifica o programa IDAMS desejado; seguindo o comando $FILES, o arquivo Dados e o arquivo Dicionário associado são especificados; o comando $RECODE seguido por declarações Recode (aqui a recodificação é utilizada para agrupar anos de educação em 4 grupos); o comando $SETUP seguido por parâmetros para a operação (nesse caso, requerendo distribuições de freqüência univariadas) são dados (de acordo com as regras do programa TABLES).

Clique em Arquivo/Salvar e salve o arquivo de setup no arquivo "demog1.set".



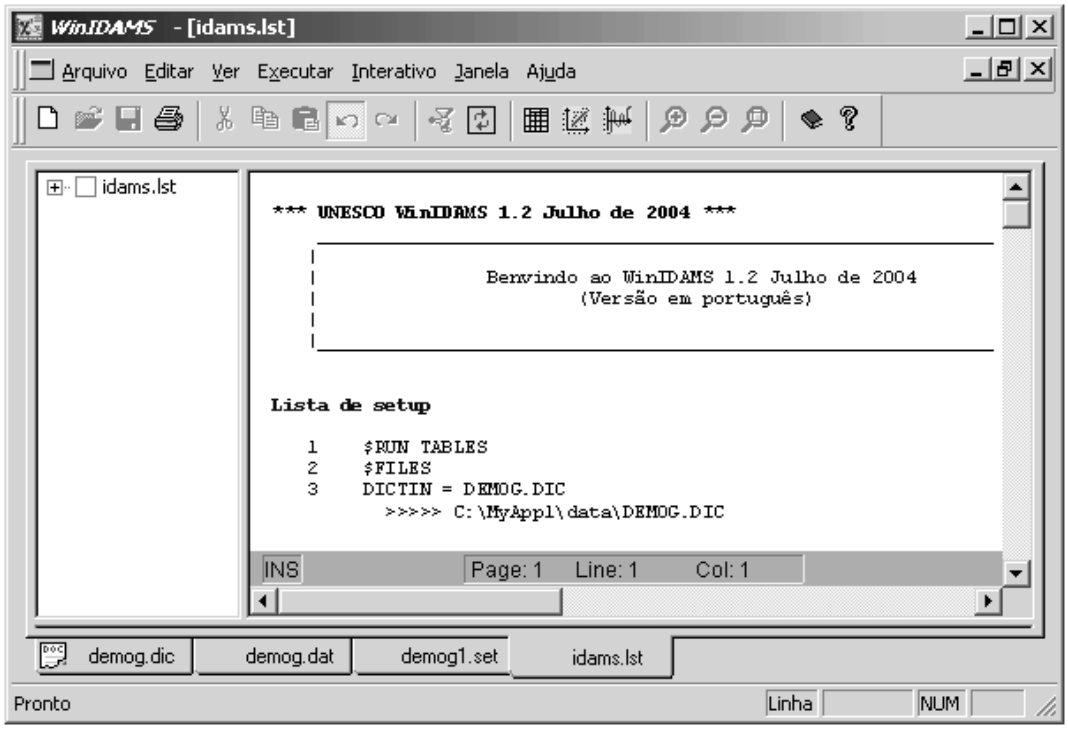
7.6 Execute o Setup

- De dentro da janela Setup, clique em Executar/Setup corrente. O setup corrente é salvo em um arquivo temporário e executado. Um diálogo aparece durante a execução e desaparece se a execução é feita com sucesso.

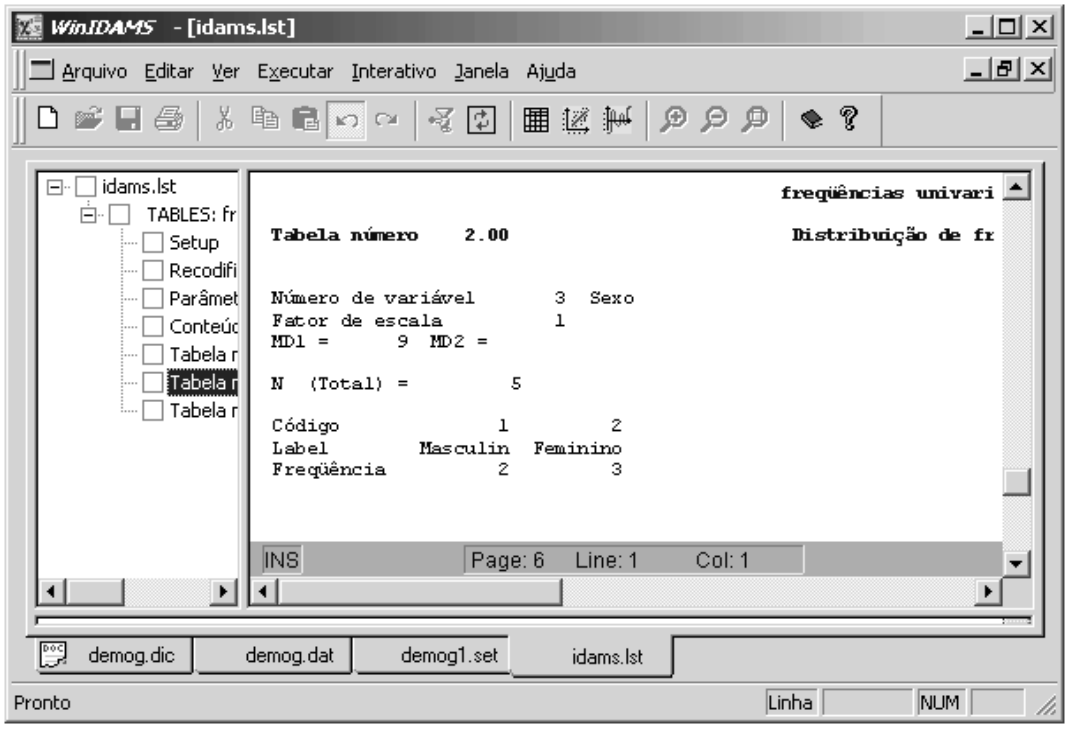
- Os resultados são, por default, escritos em um arquivo "idams.lst". Ele pode ser modificado ao se adicionar uma linha PRINT abaixo de $FILES para dar o nome do arquivo Resultados, e.g. "print=a:demog1.lst" para guardar os resultados em um arquivo de disquete.

7.7 Rever Resultados e Modificar o Setup

- O arquivo Resultados é carregado automaticamente quando a execução é terminada.



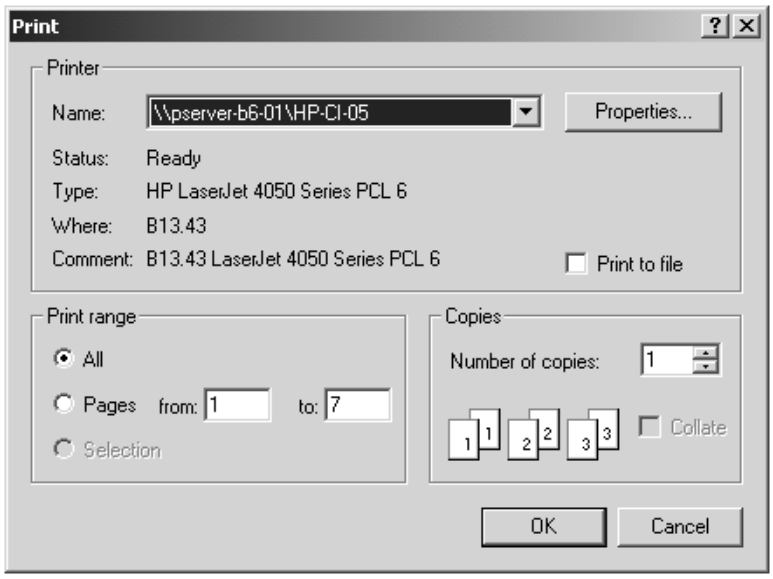
- O índice fornecido pela janela da esquerda permite a localização rápida de partes dos resultados. Abra-o clicando "idams.lst" e empurrando o botão com um asterisco no pad numérico. Então, clique no elemento que você deseja ver.



- Se você quiser mudar algo no arquivo de setup enquanto estiver revendo os resultados, clique na etiqueta "demog1.set" e faça as modificações exigidas. Pressione Ctrl/E para executar.

7.8 Imprima os Resultados

- Selecione Arquivo/Imprimir.



- Selecione as páginas que você deseja imprimir e clique em OK.

Capítulo 8

Arquivos e Folders

8.1 Arquivos em WinIDAMS

Arquivos do Usuário

Eles são criados pelo usuário com ajuda de ferramentas fornecidas pela Interface do Usuário do WinIDAMS, ou eles são produzidos por um procedimento do IDAMS como um produto final ou saída para posterior processamento. Todos os arquivos do usuário em IDAMS são arquivos de texto ASCII. São permitidos caracteres de tabulação; eles são automaticamente convertidos no número correto de espaços em branco. Extensões de arquivo padrões são usadas pela Interface para reconhecimento do tipo de arquivo.

- ***Arquivo Dados (*.dat).*** Qualquer arquivo de dados pode ser introduzido nos programas IDAMS desde que cado caso esteja contido em um número igual de registros de formato fixo. Contudo, se um arquivo Dados é usado pela Interface de Usuário do WinIDAMS, então há apenas um registro por caso.

  Registros podem ser de comprimento variável, com um máximo de 4096 caracteres por caso. Se o primeiro registro no arquivo não é o mais longo, então o comprimento máximo de registro (RECL) deve ser fornecido nas respectivas especificações de arquivo. Arquivos de dados produzidos por programas do IDAMS possuem registros de formato fixo sem caracteres de tabulação. Não há geralmente limite no número de casos que pode ser introduzido em um programa IDAMS.

- ***Arquivo Dicionário (*.dic).*** O dicionário é usado para descrever as variáveis nos dados. Ele pode, no mínimo, descrever apenas as variáveis que estão sendo utilizadas por um programa particular em execução, mas ele pode também descrever todas as variáveis em cada registro de dados. O comprimento de registro é variável, mas o comprimento máximo é de 80. Se um dicionário é produzido por um programa IDAMS, então o comprimento de registro é fixo (80 caracteres) sem caracteres de tabulação.

  O dicionário pode ser preparado sem se saber seu formato interno, na janela Dicionário da Interface do Usuário. Alternativamente, ele pode ser preparado usando o Editor Geral e seguindo o formato dado no capítulo "Dados em IDAMS".

- ***Arquivo Matriz (*.mat).*** As matrizes do IDAMS para armazenamento de várias estatísticas possuem registros de comprimento fixo (80 characters) sem caracteres de tabulação.

- ***Arquivo Setup (*.set).*** Esse arquivo é usado para armazenar comandos do IDAMS, especificações de arquivo, declarações de controle de programa e declarações de Recode (se houver). O arquivo Setup pode ser preparado na janela Setup da Interface do Usuário. O comprimento do registro é variável, apesar de o máximo ser de 255 caracteres.

- ***Arquivo Resultados (*.lst).*** IDAMS normalmente escreve os resultados em um arquivo. O conteúdo desse arquivo pode ser, então, revisto antes mesmo da impressão.

Nota: Com a finalidade de facilitar o trabalho com o WinIDAMS, é recomendável usar um nome comum para os arquivos Dados e Dicionário, como também um mesmo nome para os arquivos de Setup e Resultados.

Os arquivos de usuário são especificados no arquivo Setup seguindo o comando $FILES (ver o capítulo "O Arquivo Setup do IDAMS" para uma descrição detalhada).

Arquivos de Sistema

Normalmente, arquivos de sistema não são acessados diretamente pelo usuário. Eles são criados durante o processo de instalação (arquivos de sistema permanentes), durante a customização de uma aplicação (arquivos Aplicação) ou durante execução de procedimentos do WinIDAMS (arquivos temporários de trabalho).

- ***Arquivos de sistema permanentes.*** Eles incluem os arquivos de programas executáveis, arquivos dll, arquivos de parâmetros do sistema, arquivo com o manual on-line (em formato de HTML Help), e arquivos protótipos de setup.
- ***Arquivos de controle de sistema.***
  - *Idams.def :* definições de arquivo default fornecendo conexões entre os nomes de arquivos lógico e físicos para os arquivos do usuário e arquivos temporários de trabalho.
  - *<application_nome>.app :* um arquivo por aplicação contendo atalhos para o folder Dados, folder Trabalho e folder Temporário.
  - *lastapp.ini :* arquivo contendo o nome da última aplicação utilizada.
  - *graphid.ini :* arranjos de configuração para o componente GraphID.
  - *tml.ini :* arranjos de configuração para o componente TimeSID.
- ***Arquivos temporários de trabalho.*** Eles não devem preocupar o usuário, pois eles são definidos e removidos automaticamente. Eles possuem extensões de nome de arquivo .tmp e .tra.

8.2 Folders em WinIDAMS

Arquivos utilizados no WinIDAMS são armazenados nos seguintes folders:

- arquivos de sistema no *folder Sistema*,
- arquivos Aplicação no *folder Aplicação*,
- arquivos Dados, Dicionário e Matriz no *folder Dados*,
- arquivos Setup e Resultados no *folder Trabalho*, e
- arquivos temporários de trabalho no *folder Temporário* e *folder Transposto*.

Cinco folders, obrigatórios para a aplicação default, devem sempre estar presentes no folder `<system_dir>`. Eles são definidos e criados inicialmente durante o processo de instalação. Então, quando o WinIDAMS é iniciado e qualquer dos folders está faltando, ele é automaticamente recriado.

Folder Aplicação	`<system_dir>\appl`
Folder Dados	`<system_dir>\data`
Folder Temporário	`<system_dir>\temp`
Folder Transposto	`<system_dir>\trans`
Folder Trabalho	`<system_dir>\work`

onde `<system_dir>` é o nome do folder Sistema fixado durante a instalação.

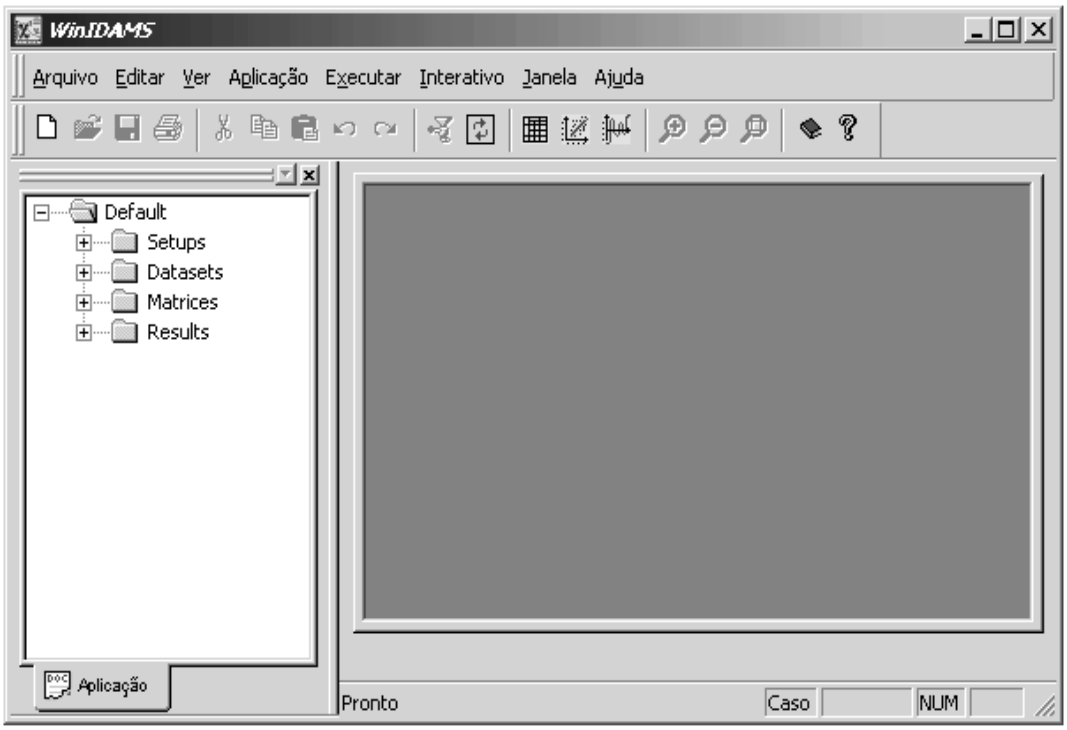
Para maiores detalhes em como os programas do IDAMS utilizam os atalhos definidos na aplicação, ver seção “Customização do Ambiente para uma Aplicação” no capítulo “Interface do Usuário”.

Capítulo 9

Interface do Usuário

9.1 Conceito Geral

A Interface do Usuário do WinIDAMS é uma interface de múltiplos documentos. Ela pode exibir e permitir que se trabalhe simultaneamente com diferentes tipos de documenos como Dicionário, Dados, Setup, Resultados e qualquer documento de texto em janelas separadas. Além disso, ela dá acesso à execução de setups do IDAMS e a componentes de análise de dados interativos, como: Tabelas multidimensionais, Exploração gráfica de dados e Análise de séries temporais de qualquer janela de documento. A janela Principal do WinIDAMS contém:



- ***a barra de menu*** para abrir os menus drop-down com comandos ou opções do WinIDAMS,
- ***a barra de ferramentas*** para escolher comandos rapidamente,
- ***a barra de status*** para exibir informação sobre o documento ativo ou comando/opção focalizada,
- ***a janela Aplicação,*** localizada no canto esquerdo, para exibir o nome da aplicação ativa, folders e documentos para essa aplicação,
- ***as janelas de documento*** para exibir documentos diferentes do WinIDAMS.

A barra de menu e a barra de ferramentas possuem conteúdos fixos, dependentes do documento. Os menus comuns são descritos baixo, enquanto menus dependentes do tipo de documento são descritos nas seções relevantes.

9.2 Menus Comuns a Todas Janelas do WinIDAMS

A barra de menu principal contém sempre os sete seguintes menus: Arquivo, Editar, Ver, Executar, Interativo, Janela e Ajuda.

Arquivo

Novo	Chama a caixa de diálogo para selecionar o tipo de documento a ser criado, e para dar o seu nome e localização.
Abrir	Depois de escolher o tipo de documento, chama a caixa de diálogo para selecionar o documento a ser aberto.
Fechar	Fecha a janela ativa.
Salvar	Salva o documento exibido na janela ativa.
Salvar como	Chama a caixa de diálogo para salvar o documento na janela ativa.
Configurar impressora	Chama a caixa de diálogo para modificar as opções de impressão e da impressora.
Visão preliminar	Exibe o documento ativo como ele será impresso.
Imprimir	Chama a caixa de diálogo para imprimir o conteúdo do documento exibido no espaço/janela ativa. Note que as partes escondidas do documento não são impressas.
Sair	Termina a sessão do WinIDAMS.

O menu pode também conter uma lista de até 7 documentos recentemente abertos, i.e. documentos usados em sessões prévias do WinIDAMS.

Editar

A disponibilidade e, às vezes, o título de alguns comandos nesse menu podem ser diferentes em janelas diferentes.

Desfazer	Cancela a última ação.
Refazer	Executa novamente a última ação cancelada.
Cortar	Move a seleção para o Clipboard.
Copiar	Copia a seleção para o Clipboard.
Colar	Copia o conteúdo do Clipboard no lugar onde o cursor estiver posicionado.
Encontrar	Inicia o mecanismo de busca do Windows.
Repor	Inicia o mecanismo de substituição do Windows.
Encontrar próximo	Procura pela próxima aparição de uma string de caracteres exibida na caixa de diálogo Encontrar.

Note que nas janelas Texto e Resultados, as ações de busca/substituição são ativadas pelos comandos Procurar, Procurar adiante, Procurar atrás e Repor.

Ver

Barra de ferramentas	Barra de ferramentas exibe/oculta.
Barra de status	Barra de status exibe/oculta.
Aplicação	Janela de aplicação exibe/oculta.
Mostrar tela completa	Exibe a janela ativa em tela cheia. Clique o ícone Fechar tela completa no canto superior esquerdo ou pressione Esc para voltar à tela anterior.

Executar

Com exceção da janela Setup, o menu possui apenas um comando, Selecionar Setup, para selecionar um arquivo com o setup a ser executado.

Interativo

Através desse menu, três componentes para análise interativa podem ser acessados:

Tabelas multidimensionais

Exploração gráfica de dados

Análise de séries temporais

Ver capítulos relevantes para uma descrição detalhada de cada componente.

Janela

O menu contém a lista de janelas abertas e comandos padrões do Windows para organizá-las.

Ajuda

Manual do WinIDAMS	Propicia acesso ao Manual de Referência do WinIDAMS.
Sobre o WinIDAMS	Exibe informação sobre a versão e direitos autorais do WinIDAMS e um link para acessar a Web page do IDAMS nasede da UNESCO.

9.3 Customização do Ambiente para uma Aplicação

Nomes do folder Dados, folder Trabalho e folder Temporário podem ser definidos pelo usuário e salvos em um arquivo Aplicação com o nome da aplicação como nome do arquivo. O nome da última aplicação utilizada é salvo pelo sistema e as configurações definidas para essa aplicação são carregadas no início da sessão seguinte. Essas configurações podem ser mudadas a qualquer tempo durante a sessão de trabalho ao selecionar/criar e ativar outra aplicação.

Como pelo menos um arquivo Aplicação é necessário para a utilização do WinIDAMS, uma aplicação standard chamada "Default" é fornecida e será ativada quando você iniciar WinIDAMS pela primeira vez depois da instalação. As configurações default definidas são as seguintes:

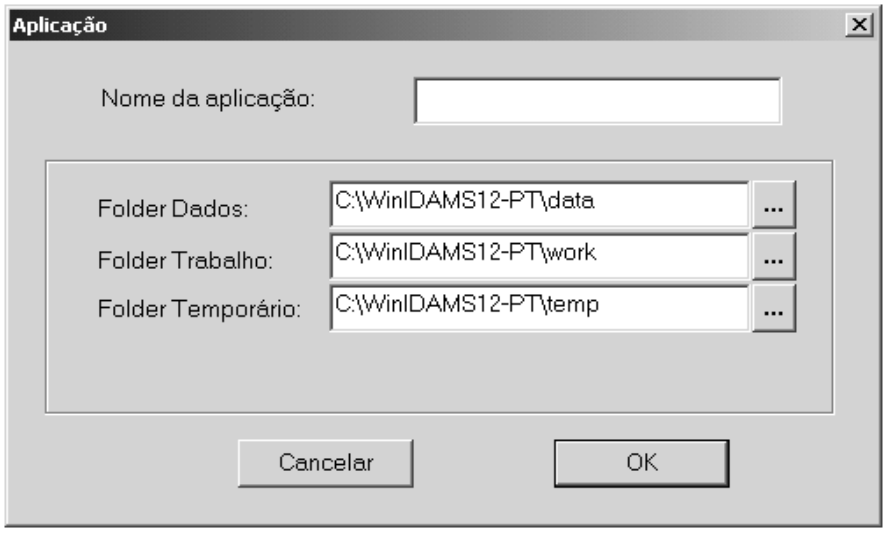
Folder Dados	`<system_dir>\data`
Folder Trabalho	`<system_dir>\work`
Folder Temporário	`<system_dir>\temp`

onde `<system_dir>` é o nome do folder Sistema fixado durante a instalação. Essa aplicação (armazenada no arquivo Default.app) não deve ser deleteda nem modificada pelo usuário.

Arquivos de aplicação (exceto Default.app) podem ser criados, modificados ou deletados pelo usuário através do menu **Aplicação** na janela Principal do WinIDAMS. Ela contém os seguintes comandos:

Nova	Chama a caixa de diálogo para criar uma nova aplicação.
Abrir	Chama a caixa de diálogo para selecionar o arquivo contendo detalhes sobre a aplicação a ser aberta.
Mostrar	Chama a caixa de diálogo para selecionar o arquivo de aplicação e exibe as configurações da aplicação.
Fechar	Fecha a aplicação ativa e abre a aplicação Default.
Atualizar	Recria a árvore da aplicação corrente.

Criando uma nova aplicação. A seleção do comando de menu Aplicação/Nova resulta em uma caixa de diálogo para entrar o nome da nova aplicação e os nomes dos folders Dados, Trabalho e Temporário. Exceto para o campo do nome da aplicação que está vazio, todos os outros campos contêm valores default extraídos da aplicação Default. Você pode digitar o nome do atalho diretamente ou selecioná-lo ao mover o highlight até o nome desejado na árvore de folders exibida.



Pressione o botão OK para salvar a aplicação. Pressionando Cancelar cancela-se a criação de uma nova aplicação e retorna-se para a janela Principal do WinIDAMS com as configurações exibidas previamente.

Abrindo uma aplicação. O comando do menu Aplicação/Abrir chama a caixa de diálogo para selecionar um arquivo de aplicação a ser aberto e fornece uma lista de aplicações existentes no folder Aplicação. Clicando nos nomes dos arquivos requeridos ativa-se as configurações para essa aplicação.

Modificando uma aplicação. Para modificar uma aplicação, primeiro abra-a e então mude os valores da mesma maneira que ao criar uma aplicação.

Exibindo as configurações para uma aplicação. Use o comando do menu Aplicação/Mostrar para chamar a caixa de diálogo e clique no nome do arquivo requerido.

Para mostar as configurações para a aplicação ativa, dê um clique duplo no seu nome na janela Aplicação.

Deletando uma aplicação. Isso pode ser feito deletando-se o arquivo correspondente. Use o comando do menu Aplicação/Abrir para obter uma lista de arquivos Aplicação, selecione o arquivo para deletar e use o botão da direita para acessar o comando Delete do Windows. O arquivo Default.app não deve ser deletado.

Reconfigurando os defaults do WinIDAMS. Para substituir uma aplicação em exibição pela aplicação default você pode fechá-la usando comando do menu Aplicação/Fechar, ou selecioná-la e abrir o arquivo Default.app.

Fechando uma aplicação ativa. Use o comando do Aplicação/Fechar. A aplicação default se torna ativa.

Programs do IDAMS usam os atalhos definidos na aplicação para prefixar qualquer nome de arquivo que não se inicie por "`<drive>:\...`" ou "`\...`"

- O atalho do folder Dados é prefixado para todos os nomes de arquivo em declarações com ddnames DICT..., DATA..., ou FTnn referindo-se a matrizes.
- O atalho do folder Trabalho é prefixado para nomes de arquivos em declarações com ddnames PRINT ou FT06.
- O atalho do folder Temporário é prefixado para arquivos temporários.

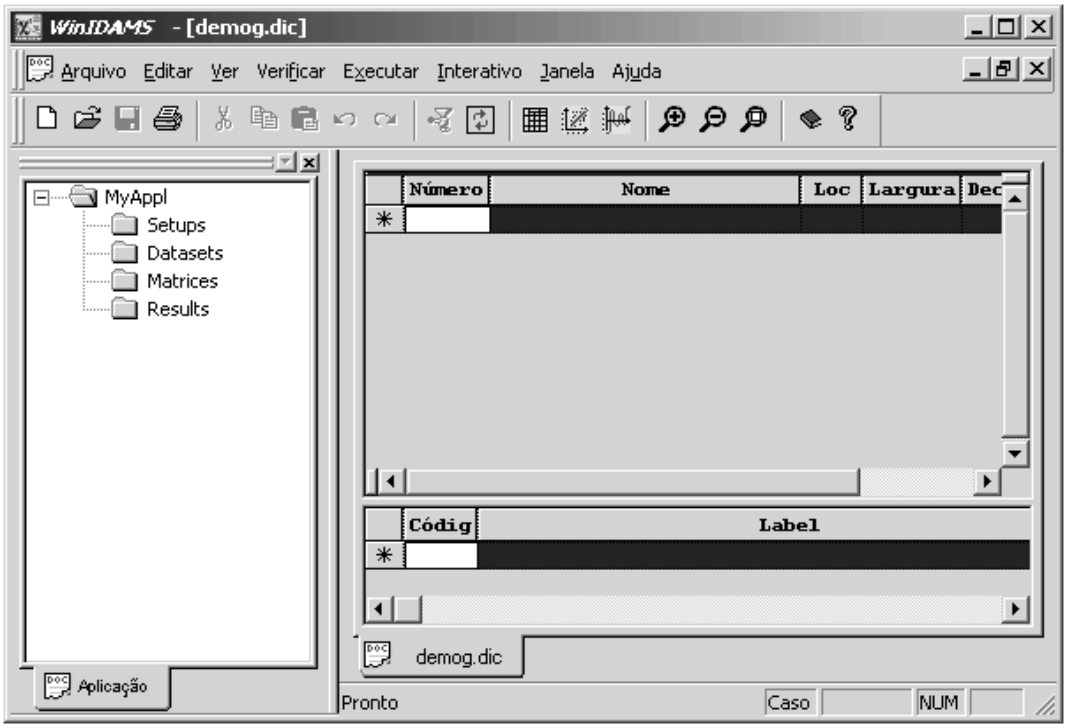
Exemplos:

```
Folder Dados:                          c:\MyStudy\students\data
Especificação no setup:                dictin=students2004.dic
Nome completo de arquivo dicionário:   c:\MyStudy\students\data\students2004.dic
```

9.4 Criando/Renovando/Exibindo Arquivos Dicionário

A janela Dicionário para criar, renovar ou exibir um dicionário IDAMS, é chamada quando:

- você cria um novo arquivo Dicionário (o comando do menu Arquivo/Novo/IDAMS Dictionary file ou o botão Novo da barra de ferramentas),
- você abre um arquivo Dicionário (com extensão .dic) exibido na janela Aplicação (dê um clique duplo no nome do arquivo requerido na lista "Datasets"),
- você abre um arquivo Dicionário (com qualquer extensão) que não está na janela Aplicação (o comando do menu Arquivo/Abrir/Dicionário ou o botão Abrir da barra de ferramentas).



A janela oferece dois espaços: um para as definições de variáveis (espaço Variáveis) e outro para os códigos e labels de códigos da variável corrente (espaço Códigos). Uma linha azul no topo de cada espaço indica qual espaço está ativo.

Os cabeçalhos das colunas no espaço das Variáveis possuem o seguinte significado:

Número	Número da variável.
Nome	Nome da variável.
Loc, Largura	Localização de início e largura de campo da variável no arquivo Dados.
Dec	Número de casas decimais; espaço em branco implica nenhuma casa decimal.
Tipo	Tipo de variável (N = numérica, A = alfabética).
Md1	Primeiro código de dados perdidos para variáveis numéricas.
Md2	Segundo código de dados perdidos para variáveis numéricas.
Refe	Número de referência.
IdEs	ID do estudo.

Para maiores detalhes, ver a seção "O Dicionário do IDAMS" no capítulo "Dados em IDAMS". Note que apenas dicionários descrevendo dados com um registro por caso podem ser criados, renovados ou exibidos usando a janela Dicionário.

Modificando a aparência da janela. A aparência de cada espaço na janela pode ser modificado separadamente e a mudança se aplica apenas para o espaço ativo.

As seguinte possibilidades de mudança estão diponíveis para cada espaço de janela:

- *Aumentando o tamanho da fonte* - use o botão Zoom In da barra de ferramentas.
- *Diminuindo o tamanho da fonte* - use o botão Zoom Out da barra de ferramentas.
- *Recuperando o tamanho default da fonte* - use o botão 100% da barra de ferramentas.
- *Aumentando/Diminuindo a largura da coluna* - coloque o cursor do mouse na linha que separa duas colunas no cabeçalho da coluna até o cursor se torna uma barra vertical com duas setas e mova-o para a direita/esquerda segurando o botão esquerdo do mouse.

Os espaços das variáveis pode ser ainda mais modificado da seguinte maneira:

- *Aumentando/Diminuindo a altura das linhas* - coloque o cursor do mouse na linha que separa duas linhas no início da linha até que o cursor se torne uma barra horizontal com duas setas e mova-o para cima/baixo segurando o botão esquerdo do mouse.

Definindo uma variável. Coloque o cursor no espaço Variáveis, preencha o número de variáveis (pelo menos uma é mandatório, variáveis subseqüêntes serão numeradas adicionando-se o valor 1), nome (opcional), localização (se não colocado, será designado o valor de 1 para a primeira variável e para variáveis subseqüêntes, a localização será calculada pela adição da largura da variável precedente) e largura (mandatório). Outros campos possuem valores default (que você pode aceitar ou modificar) ou eles são opcionais e podem ser deixados em branco. Pressione Enter ou Tab para aceitar um valor em um campo e mover para o próximo campo, ou Shift/Tab para mover para um campo anterior. Note que desde que um pequeno lápis apareça no início da linha, a linha não é salva. Pressione Enter para aceitar a definição completa da variável. Um asterisco no cabeçalho da linha indica que essa é a próxima linha e você pode entrar uma nova descrição de variável.

Definindo os códigos e as labels dos códigos para uma variável. Mude para o espaço de janela Códigos a preencha os campos de código e label de código. Preencha o valor de código, então pressione Enter ou Tab e entre com a label do código, então Enter ou Tab para aceitar a linha e vá para a próxima linha. Quando todos os códigos e labels tiverem sido definidas, mude para o espaço Variáveis para continuar com outra definição de variável.

Modificando o campo no espaço Variáveis ou no espaço Códigos. Clique no campo e entre o novo valor (entrando o primeiro caracter do novo valor limpa o campo). Depois de dar um clique duplo em um campo, seu valor corrente pode ser modificado em parte. A tecla Esc pode ser usada para recuperar valores anteriores.

Operações de edição podem ser executadas em uma linha ou bloco de linhas. Para marcar uma linha, clique em qualquer campo dessa linha. Um triângulo aparece no cabeçalho da linha e a linha é colorida de azul escuro. Para marcar um bloco de linhas, coloque o cursor do mouse no cabeçalho da linha onde você

quer iniciar a marcação e clique o botão esquerdo do mouse. A linha se torna amarela, indicando que está ativa. Então movemente o cursor do mouse acima ou abaixo para a linha onde você quer finalizar a marcação e clique o botão esquerdo do mouse mantendo a tecla Shift. Linhas marcadas se tornam azul escuro, a cor amarela mostra a linha ativa.

Você pode Cortar, Copiar e Colar linha(s) marcada(s) usando os comandos do Editar, botões de barra de ferramentas equivalentes ou teclas de shortcut Ctrl/X, Ctrl/C ou Ctrl/V, respectivamente.

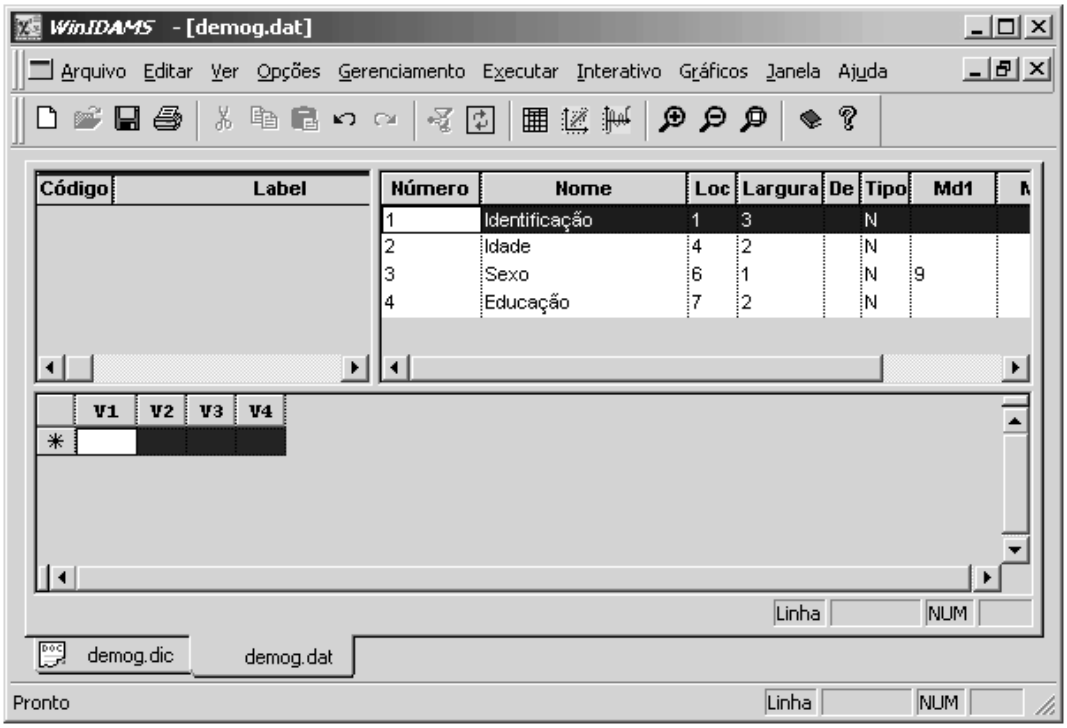
Usando o botão direito do mouse você pode Inserir antes, Inserir depois, Deletar ou Limpar a linha ativa (quando um bloco de linhas é marcada).

Detectando erros em um dicionário. Use o comando de menu Verificar/Validade. Erros são mostrados um por um e podem ser corrigidos quando eles todos são exibidos. Além do mais, Interface tenta prevenir você de salvar dicionários com erros. Também, quando você abre um dicionário com erros, a presença deles é sinalizada antes do dicionário ser realmente aberto.

9.5 Criando/Renovando/Exibindo Arquivos Dados

A janela Dados é usada para criar, renovar ou exibir um arquivo Dados do IDAMS. Note que o arquivo Dados correspondente já deve ter sido construído e que apenas arquivos Dados com um registro por caso pode ser criado, renovado ou exibido usando a janela Dados. Essa janela é chamada quando:

- você cria um arquivo Dados (o comando de menu Arquivo/Novo/IDAMS Data file ou o botão Novo da barra de ferramentas),
- você abre um arquivo Dados (com extensão .dat) exibido na janela Aplicação (dê um clique duplo no nome do arquivo requerido na lista “Datasets”),
- você abre um arquivo Dados (com qualquer extensão) que não está na janela Aplicação (o comando de menu Arquivo/Abrir/Dados ou o botão Abrir da barra de ferramentas).


WinIDAMS - [demog.dat]
Arquivo Editar Ver Opções Gerenciamento Executar Interativo Gráficos Janela Ajuda
Código Label
Número Nome Loc Largura De Tipo Md1
1 Identificação 1 3 N
2 Idade 4 2 N
3 Sexo 6 1 N 9
4 Educação 7 2 N
V1 V2 V3 V4
Linha NUM
demog.dic demog.dat
Pronto Linha NUM

A janela é dividida em três partes: uma mostrando os códigos e as labels de código da variável corrente (espaço Códigos), a segunda mostrando as definições das variáveis (espaço Variáveis) e a terceira oferecendo espaço para entrada/modificação de dados (espaço Dados). Apenas o espaço Dados pode ser editado. Os outros dois espaços apenas exibem as informações relevantes. Uma linha azul no topo de cada espaço indica

qual espaço está ativo. Os espaços são sincronizados, i.e. seleção do campo de uma variável no espaço Dados evidencia a descrição da variável correspondente, e seleção de um campo do espaço Variáveis mostra o valor da variável correspondente no caso corrente. Para a variável selecionada, códigos e labels de código (se houver) são sempre exibidos.

Mudando a aparência do espaço. A aparência de cada espaço pode ser mudada separadamente e a mudança se aplica exclusivamente ao espaço ativo.

As seguintes possibilidades de modificação estão disponíveis em todos os espaços:

- *Aumentando o tamanho da fonte* - use o comando de menu Ver/Zoom In ou o botão Zoom In da barra de ferramentas.
- *Diminuindo o tamanho da fonte* - use o comando de Ver/Zoom Out ou o botão Zoom Out da barra de ferramentas.
- *Retornando ao tamanho de fonte default* - use comando de menu Ver/100% ou o botão 100% da barra de ferramentas.
- *Aumentando/Diminuindo a largura da coluna* - coloque o cursor do mouse na linha que separa duas colunas, bem no início até que o cursor se torne uma barra vertical com duas flechas e movimente-o para a direita/esquerda mantendo o botão esquerdo do mouse apertado.

O espaço Dados pode ser modificado ainda mais da seguinte maneira:

- *Aumentando/Diminuindo a altura das linhas* - coloque o cursor do mouse na linha que separa duas linhas, bem no início da linha até que o cursor se torne uma barra horizontal com duas flechas, e então movemente para cima/baixo mantendo o botão esquerdo do mouse apertado.
- *Posicionando colunas no início* - marque as colunas desejadas e use o comando de menu Ver/Congelar coluna (use o comando de menu Ver/Descongelar colunas para colocá-las de volta).
- *Exibindo dados em espaços múltiplos* - use o comando de menu Janela/Dividir. Você obterá uma cruz para determinar o tamanho de quatro espaços. Esse tamanho pode ser mudado mais tarde usando-se a técnica padrão do Windows. Os seus dados completos são exibidos quatro vezes. A quebra horizontal pode ser removida através de um clique duplo na linha horizontal, a quebra vertical pode ser removida através de um clique duplo na linha vertical, e todas as quebras podem ser removidas através de um clique duplo no centro.

Entrando com um novo caso. Clique o primeiro campo em uma linha vazia e comece a entrar os valores dos dados. Pressione Enter ou Tab para aceitar um valor de dados para a variável e mova para a próxima variável, ou Shift/Tab para mover-se para a variável anterior. Note que desde que um pequeno lápis apareça no início da linha, o caso ainda não está salvo. Pressionando Enter na última variável salva o caso e move o cursor para o início da próxima linha. Uma nova linha pode ser inserida antes ou depois da linha evidenciada (clique no botão direito do mouse), ou pode ser adicionada no final do arquivo (linha com asterisco no início da linha).

A entrada de dados pode ser facilitada tomando vantagem de duas opções dadas no menu Opções:

Checagem de código checa os valores de dados durante a entrada de dados em relação aos códigos definidos no dicionário, que são os únicos códigos considerados válidos.

Salto automático move o cursor automaticamente para o próximo campo quando dígitos em número suficiente tenham sido introduzidos para preencher o campo. Se não selecionado, você deve pressionar Enter ou Tab para movimentar-se para o próximo campo.

Modificando o valor de uma variável. Clique o campo da variável e entre com o novo valor (entrando o primeiro caracter do novo valor limpa o campo). Um clique duplo no campo de uma variável pode ser usado para modificar parte do valor corrente. A tecla Esc pode ser usada para recuperar o valor prévio.

Copiando o valor de uma variável em um outro campo. Clique no campo da variável e copie o seu conteúdo para o Clipboard (comando Editar/Copiar, Ctrl/C ou botão Copiar na barra de ferramentas).

Então clique o campo requerido e passe o valor (comando Editar/Colar, Ctrl/V ou botão Colar da barra de ferramentas). O comando Editar/Desfazer caso pode ser utilizado para recuperar o valor prévio.

Operações de edição em uma linha ou em um bloco de linhas pode ser realizado da mesma maneira que em uma janela Dicionário. Para marcar uma linha, clique em qualquer campo dessa linha. Um triângulo aparece no início da linha e a linha é colorida de azul escuro. Para marcar um bloco de linhas, coloque o cursor do mouse no início da linha onde você deseja iniciar a marcação e clique o botão esquerdo do mouse. A linha se torna amarela, indicando que está ativa. Então, mova o cursor para cima ou baixo até a linha onde você deseja marcar e clique o botão esquerdo do mouse, mantendo a tecla Shift apertada. As linhas marcadas se tornam azul escuras, e a cor amarela mostra a linha ativa.

Você pode Cortar, Copiar e Colar linha(s) marcada(s) usando os comandos Editar, botões equivalentes na barra de ferramentas ou teclas de atalho Ctrl/X, Ctrl/C e Ctrl/V, respectivamente.

Usando o botão direito do mouse você pode Inserir antes, Inserir depois, Deletar ou Copiar a linha ativa (mesmo quando um bloco de linhas está marcado).

Dois comandos de gerenciamento de dados são oferecidos no menu Gerenciamento para permitir a verificação e classificação de dados:

Checar códigos checa os valores dos dados para todos os casos no arquivo Dados em relação aos códigos definidos no dicionário, sendo estes os únicos códigos considerados válidos. Ao final da verificação, uma menssagem mostrando o número de erros encontrado é exibida e você é convidado a corrigí-los, um a um, usando a caixa de diálogo de correção de dados. Essa caixa fornece número seqüêncial de casos, número e nome da variável, valor de código inválido e uma lista drop-down de códigos válidos como definidos no dicionário.

Classificar chama uma caixa de diálogo de classificação para especificar até 3 variáveis de classificação e a ordem correspondente de classificação para cada uma delas. Depois de clicar OK, o arquivo classificado aparce na janela Dados.

Classificar dados de **uma** variável (uma coluna) pode também ser feito dando uma clique duplo no número da variável na início do espaço Dados. Um duplo clique classifica os casos em ordem crescente. Para obter uma classificação em ordem decresente, repita o clique duplo.

Dois tipos de gráficos são sugeridos para uma variável no menu Gráficos.

Gráfico de barras fornece as freqüências ou percentuais para categorias de variáveis qualitativas. Para variáveis quantitativas, o usuário define o número de barras (NB), em ambos os lados da média (M) e o coeficiente (C) para calcular a largura da barra (classe). A largura da barra (BW) é igual ao valor do desvio-padrão (STD) multiplicado pelo coeficiente (BW=C*STD). As barras são construídas usando os valores M-NB*BW, ..., M-2BW, M-BW, M, M+BW, M+2BW, ..., M+NB*BW. A altura de um retângulo = (freqüência relativa da classe)/(largura da classe). Além disso, a curva de distribuição normal tendo a média e o desvio-padrão calculados pode ser projetada para variáveis quantitativas.

Histograma, recomendado para variáveis quantitativas, fornece um histograma baseado em freqüências ou em percentuais com o número de bins especificado pelo usuário.

Gráficos para variáveis quantitativas contém também estatísticas univariadas para as variáveis projetadas como: média, desvio-padrão, variância, assimetria e curtose. Variáveis com casas decimais são multiplicadas por um fator de escala para obter valores inteiros. Nesse caso, a média, desvio-padrão e variância devem ser ajustados de acordo.

9.6 Importando Arquivos de Dados

WinIDAMS fornece uma ferramenta para importar arquivos de dados para o IDAMS diretamente através da Interface do Usuário do WinIDAMS. Essa facilidade pode ser acessada da janela Principal do WinIDAMS, da janela Dados e da janela Tabelas multidimensionais.

Três tipos de arquivos de formato livre podem ser importados:

- arquivos .txt nos quais os campos são separados por tabs,

- arquivos .csv nos quais os campos são separados por vírgulas,
- arquivos .csv nos quais os campos são separados por ponto-e-vírgula.

Infomação fornecida na primeira linha é considerada ser labels de coluna e é utilizada como nomes de variáveis no processo de construção do dicionário. Portanto, a presença de labels de coluna é mandatória na primeira linha de arquivos de entrada.

O caracter de separação é detectado na primeira linha, enquanto o caracter usado como separador decimal é detectado na segunda linha do arquivo. Portanto, a presença de caracters decimais é mandatória na segunda linha de arquivos de entrada se uma variável contém decimais.

Durante o processo de importação, o conteúdo de variáveis alfabéticas importadas pode ser convertido para códigos numéricos, mantendo os valores alfabéticos como labels de código no dicionário de IDAMS criado. Vírgulas utilizadas como separador decimal são convertidas em pontos.

A operação de Importação de Dados é ativada com o comando Arquivo/Importar, seguido pela seleção do arquivo requisitado na caixa de diálogo Abrir no arquivo padrão. O caracter de separação e o caracter usado como separador decimal são exibidos junto com os valores de todos os campos para os três primeiros casos. A leitura de dados podem então ser checada antes de iniciar-se a importação. Depois disso, você terá a seu dispor duas janelas chamadas Dados externos e Definição de variáveis, ambas em forma de planilha.

A janela Dados externos mostra apenas o conteúdo do arquivo a ser importado. Nenhuma operação de edição é permitida, exceto copiar uma seleção para o Clipboard.

A janela Definição de variáveis serve para preparar descrições de variáveis do IDAMS. Seu conteúdo inicial é fornecido por default e com base nos dados importados, mas você é livre para modificá-lo e completá-lo se necessário.

As colunas contêm as seguintes informações:

Descrição	Nome da variável.
Tipo	Tipo de variável (númerica por default). Esse é o tipo da variável de entrada. Se uma variável de entrada é alfabética e deve ser processada como numérica, opte pela recodificação (ver abaixo).
LargMáx	Largura máxima do campo de variável.
NumDec	Número de casas decimais; espaço em branco significa nenhuma casa decimal.
Md1	Primeiro código de dados perdidos para variáveis numéricas.
Md2	Segundo código de dados perdidos para variáveis numéricas.
Recodificação	Requerendo uma recodificação de variáveis alfabéticas para valores numéricos.

Para modificar as definições de variáveis, coloque o cursor dentro da janela. Então utilize as teclas de navegação ou o mouse para mover para o campo requerido e mude o conteúdo.

Use o comando de menu Construir/Dataset de IDAMS para criar arquivos Dicionário e Dados do IDAMS. Eles serão ambos colocados no folder Dados da aplicação corrente.

9.7 Exportando Arquivos Dados do IDAMS

WinIDAMS possui também uma ferramenta para exportar arquivos Dados do IDAMS através da Interface do Usuário do WinIDAMS. Isso pode ser feito na janela Dados usando o comando Arquivo/Exportar. O arquivo Dados do IDAMS exibido na janela ativa pode ser salvo em um dos três formatos livres de arquivos de dados:

- arquivos .txt nos quais os campos são separados por tabs,
- arquivos .csv nos quais os campos são separados por vírgulas,
- arquivos .csv nos quais os campos são separados por ponto-e-vírgula.

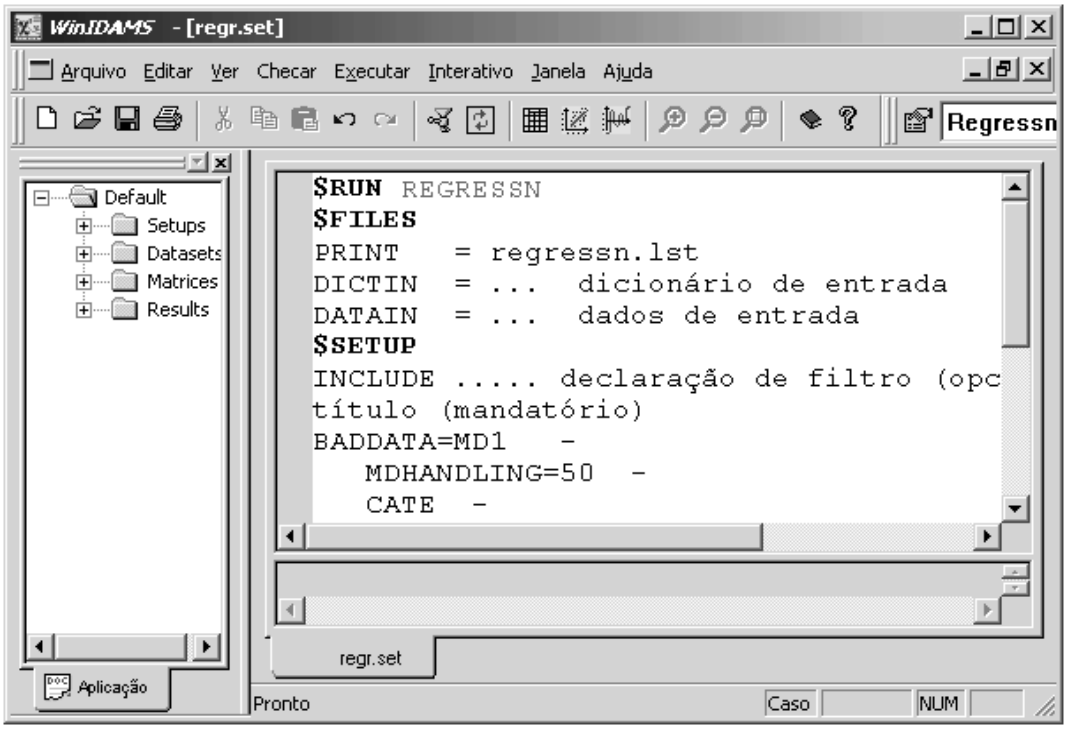
Os nomes da variáveis do arquivo Dicionário correspondente são obtidos como labels de coluna na primeira linha dos dados exportados.

Se existem labels de código para uma variável, valores de códigos numéricos podem ser opcionalmente substituídos pelos seus labels de código correspondentes no arquivo de dados de saída. Além do mais, variáveis numéricas podem ser processadas com a vírgula usada como separador de casas decimais.

9.8 Criando/Renovando/Exibindo Arquivos Setup

A janela Setup para preparar ou exibir um arquivo Setup do IDAMS é chamada quando:

- você cria um arquivo setup (o comando de menu Arquivo/Novo/IDAMS Setup file ou o botão Novo da barra de ferramentas),
- voce abre um arquivo Setup (com extensão .set) exibibido na janela Aplicação (dê um clique duplo no nome do arquivo requisitado na lista "Setups"),
- você abre um arquivo Setup (com qualquer extensão) que não está na janela Aplicação (o comando de menu Arquivo/Abrir/Setup ou o botão Abrir da barra de ferramentas).



A janela fornece dois espaços: o de cima é para preparar o arquivo Setup (espaço Setup) e o de baixo é para exibir mensagens de erro quando as declarações de filtro e Recode são checadas (espaço Mensagens). Somente o espaço Setup pode ser editado. Note que os comandos do IDAMS são exibidos em negrito e os nomes dos programs em róseo semeles forem escritos corretamente. Texto colocado em um comando $comment é exibido em verde.

Para preparar um novo setup de programa, você pode digitar todas as declarações ou você pode utilizar o protótipo de setup para o programa requerido e modificá-lo como necessário. Protótipos de setups são fornecidos para todos os programas. Eles podem ser acessados ao selecionar-se o nome do programa na lista sob o botão Prototypes da barra de ferramentas. Para copiar o protótipo para o espaço do Setup, clique no nome de programa requisitado. Para detalhes em como preparar setups, ver o capítulo "O Arquivo Setup do IDAMS" e o write-up de programa relevante.

Operações de edição podem ser executadas como qualquer editor de arquivos ASCII, i.e. você pode Cortar, Copiar e Colar qualquer seleção, utilizando os comandos Editar, botões da barra de ferramentas equivalentes ou teclas de atalho Ctrl/X, Ctrl/C e Ctrl/V respectivamente.

Dois comandos de verificação de setup são fornecidos no menu Checar para permitir a verificação de sintaxe de conjuntos de declarações de Recode e declarações de filtro:

Sintaxe de Recode ativa a verificação de sintaxe em declarações de Recode inclusos no setup. Todos os erros encontrados são reportados no espaço Mensagens dando o número de conjunto de Recode, linha da declaração com erro(s) caracter(es) causador(es) do problema de sintaxe. Um clique duplo no texto da linha com erro ou na mensagem de erro no espaço Mensagens mostra essa linha no espaço Setup com uma flecha amarela. Você pode corrigir os erros e repetir a verificação de sintaxe, antes de enviar o setup para a execução.

Sintaxe de filtro ativa a verificação de sintaxe de erros nas declarações de filtro incluídas no setup. Todos os erros encontrados são reportados no espaço Mensagens dando o número da declaração de filtro, linha da declaração errada e caracter(es) causando o problema de sintaxe. Um clique duplo no texto da linha com erro ou na mensagem de erro no espaço Mensagens mostra essa linha no espaço de Setup com uma flecha amarela.

Note que apesar da maioria dos erros de sintaxe nas declarações de filtro e de Recode puderem ser detectados e corrigidos aqui, outra verificação de sintaxe é sistematicamente executada pelo IDAMS durante a execução do setup. Também, erros de execução que não são detectados aqui, são reportados nos resultados.

9.9 Executando Setups do IDAMS

Para executar programas do IDAMS (para o qual instruções foram preparadas e salvas em um arquivo Setup), use o comando Executar/Selecionar Setup em qualquer janela de documento do WinIDAMS. Você será requisitado, através de uma caixa de diálogo padrão do Windows, a selecionar o arquivo de onde as instruções devem ser obtidas durante execução.

Se você estiver preparando suas instruções na janela Setup, você pode executar programas do Setup corrente utilizando o comando de menu Executar/Setup corrente.

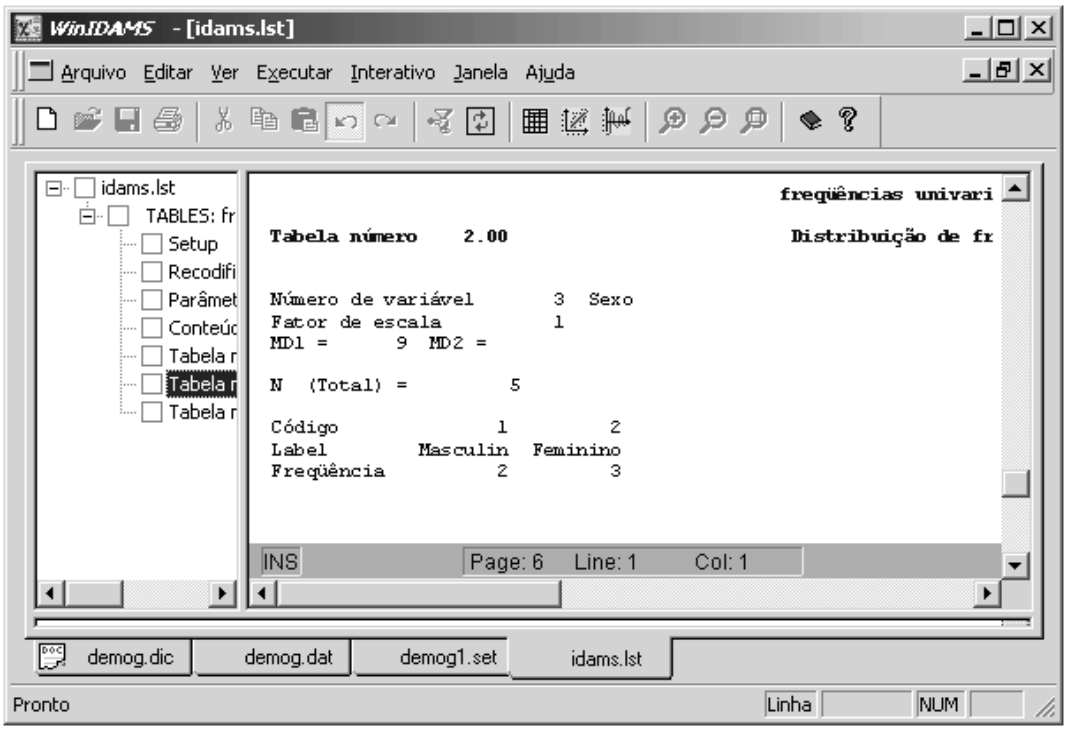
Os programas serão executados e os resultados escritos no arquivo especificado em PRINT no $FILES (o default é IDAMS.LST no folder Trabalho corrente). No final da execução, o arquivo Resultados será aberto na janela Resultados.

9.10 Manuseando Arquivos Resultados

A janela Resultados para acessar, exibir e imprimir partes selecionadas dos resultados é chamada quando:

- você abre o arquivo Resultados (com extensão .lst) exibido na janela Aplicação (dê um clique duplo no nome do arquivo requerido na lista "Results"),
- você abre um arquivo Resultados (com qualquer extensão) que não está na janela Aplicação (o comando de menu Arquivo/Abrir/Resultados ou o botão Abrir da barra de ferramentas),
- você executa setup do IDAMS; o conteúdo do arquivo Resultados é exibido automaticamente.

Navegação rápida pelos resultados é facilitada através das tabelas de conteúdo. Você pode acessar o início de resultados de um programa particular ou mesmo uma seção em particular. Além disso, o menu Editar fornece acesso a uma facilidade de busca.



A janela é dividida em três espaços: um mostrando a tabela de conteúdo (TOC) dos resultados como uma árvore, o segundo mostrando os resultados propriamente ditos e o terceiro exibindo mensagens de erro e de advertência incluídos nos resultados.

Por default, a paginação dos resultados obtidos pelos programas é retida (a opção **Modo de página** na caixa de checagem do menu **Ver** está marcada). Para tornar os resultados mais compactos, desmarque essa opção. Linhas brancas no final serão removidas de todas as páginas e quebras de página inseridas por programas serão substituídas por linhas de texto "Page break".

Para abrir/fechar rapidamente a árvore TOC, três botões no pad numérico estão disponíveis:

*	abre todos os níveis da árvore sob o nó selecionado
-	fecha todos os níveis da árvore sob o nó selecionado
+	abre um nível sob o nó selecionado.

Para visualizar uma parte particular dos resultados dê um clique duplo no seu nome na TOC.

Para localizar uma mensagem de erro ou advertência, dê um clique duplo no seu texto.

Modificação dos resultados não é permitida. Contudo, partes selecionadas (evidenciadas ou marcadas em tick-boxes na árvore TOC) ou todos os resultados podem ser copiados para o Clipboard (comando Editar/Copiar, botões Ctrl/C ou Copiar na barra de ferramentas) e passados para qualquer documento usando técnicas padrões do Windows.

Impressão do conteúdo completo ou páginas selecionadas dos resultados pode ser conseguida através do comando do menu Arquivo/Imprimir ou utilizando o botão Imprimir da barra de ferramentas. Note que a impressão é feita na orientação Paisagem, e que essa orientação não pode ser modificada.

O conteúdo do arquivo Resultados como exibido pode ser **salvo** em formato RTF ou texto usando o comando de menu Arquivo/Salvar como. Linhas em branco no final são sempre removidas. Quebras de páginas são manuseadas de acordo com a opção Modo de página.

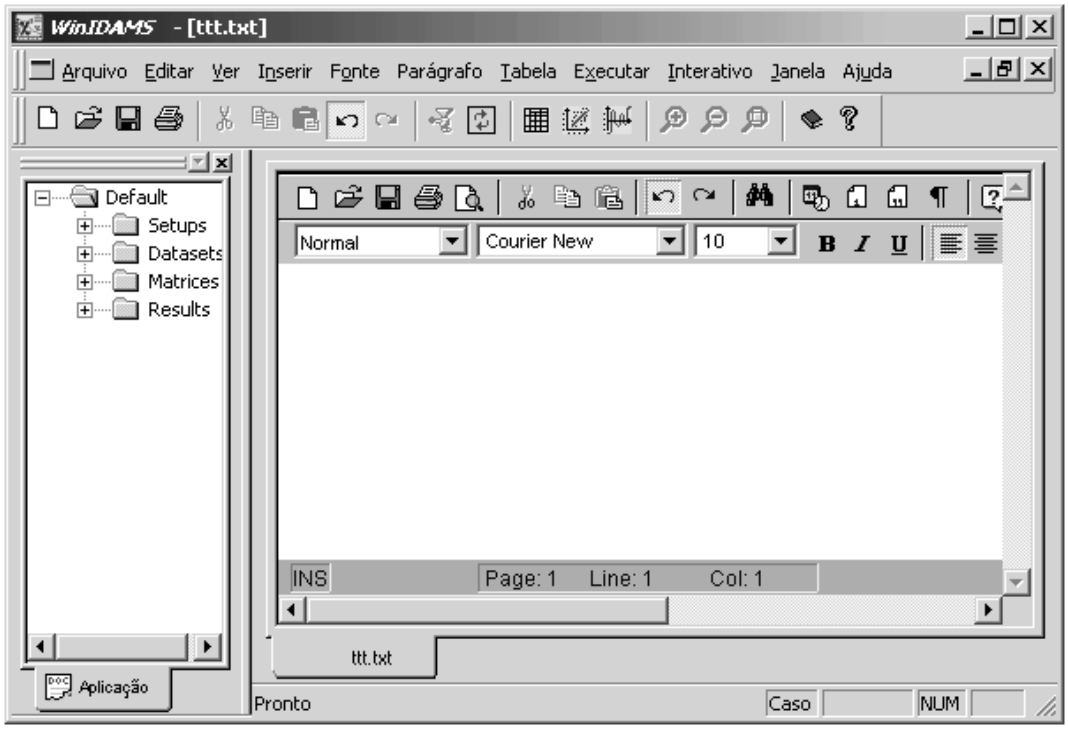
9.11 Criando/Renovando Arquivos em Formato Texto e RTF

WinIDAMS possui um Editor Geral que permite a você abrir e modificar qualquer tipo de documento em formato caracter. Contudo, sua função básica é oferecer uma facilidade para editar arquivos Texto e oferecer opções sofisticadas de formatação e edição. Manipulação de arquivos Dicionário, Dados ou Setup usando o

Editor Geral deve ser evitada, e manipulação de arquivos Matriz deve ser feita com cuidado.

A janela Texto é chamada quando:

- você cria um novo arquivo Texto (o comando de menu Arquivo/Novo/Text file or RTF file, ou o botão Novo da barra de ferramentas),
- você abre um arquivo Matriz (com extensão .mat) exibido na janela Aplicação (dê um clique duplo no nome do arquivo requerido na lista "Matrices"),
- você abre um arquivo de caracter que não está na janela Aplicação (o comando de menu Arquivo/Abrir/File Using General Editor ou o botão Abrir da barra de ferramentas).



O Editor Geral propicia um número de comandos de edição padrão que são conhecidos pelos usuários do Windows. Eles são listados abaixo mas não serão descritos em detalhe.

Inserir fornece comandos para a inserção de quebras de página e seção, figuras, objetos OLE (Object Linking & Embedding), emolduramento e desenho de objetos.

Fonte permite a você modificar a fonte e cor do texto selecionado, e a cor do pano de fundo.

Parágrafo permite ao usuário alinhar parágrafos diferenciadamente, identá-los, exibí-los em espaço duplo, e desenhar uma borda e sobras no fundo.

Tabela dá acesso a um número de comandos para inserir e manipular tabelas.

Ver contém três comandos adicionais para exibir o documento ativo em modo de página, para exibir a régua e o marcador de parágrafo.

Barra de ferramentas de formatação permite que se escolha rapidamente comandos de formatação que são usados mais freqüentemente.

Parte III

Facilidades para Gerenciamento de Dados

Capítulo 10

Agregação de Dados (AGGREG)

10.1 Descrição Geral

AGGREG agrega registros individuais (casos de dados) em grupos definidos pelo usuário e computa um sumário de estatísticas descritivas para variáveis especificadas em cada grupo. As estatísticas incluem somas, médias, variâncias, desvios-padrões, como também valores mínimos e máximos e a contagem de valores de dados perdidos. Um dataset de saída do IDAMS é criado, i.e. o arquivo de dados agrupado (agregado) descrito pelo dicionário do IDAMS; o arquivo de dados agregados contém um registro (caso) por grupo com variáveis que são o sumário para o nível do grupo de cada variável de entrada selecionada.

Fórmulas para o cálculo da média, variância e desvio-padrão podem ser encontradas na Parte "Fórmulas Estatísticas e Referências Bibliográficas", capítulo "Tabelas Univariadas e Bivariadas". Contudo, elas precisam ser ajustadas, pois os casos não são ponderados e o coeficiente N/(N-1) não é utilizado no cálculo da variância amostral e/ou desvio-padrão. Note que o sumário de estatísticas é selecionado para o conjunto inteiro de variáveis agregadas. Portanto, se houver 2 variáveis agregadas e se 3 estatísticas são selecionadas, haverá 6 variáveis computadas.

AGGREG dá condições a que o usuário mude o nível de agregação dos dados e.g. de membros individuais da família para domicílio, ou de distrito para nível regional, etc. Por exemplo, suponha que um arquivo de dados contenha registros de cada indivíduo em um domicílio e que nós quiséssemos analisar esses dados ao nível de domicílio. AGGREG permitiria-nos agregar valores de variáveis através de todos os registros individuais para cada domicílio para criar um arquivo de registros a nível de domicílio para análise posterior. Se, para ser mais específico, o arquivo de dados a nível individual continha uma variável dando a renda pessoal, AGGREG poderia criar registros ao nível de domicílio com uma variável de renda total do domicílio.

Agrupamento de dados. O usuário especifica até 20 variáveis de definições de grupo (ID) que determinam o nível de agregação do aquivo de saída. Por exemplo, se alguém quisesse agregar dados a nível individual ao nível de domicílio, uma variável identificando o domicílio seria a variável de definição de grupo. Cada vez que AGGREG lê um registro de entrada, ele checa se ocorreu alguma mudança em quaisquer variáveis ID. Quando isso é encontrado, um registro é produzido contendo o sumário de estatísticas das variáveis agregadas especificadas para o grupo de registros que acabou de ser processado.

Inserindo constantes nos registros de grupos. Constantes podem ser inseridas nos registros de grupos usando parâmetros PAD1, ... , PAD5, que especificam as chamadas variáveis pad. O valor de uma variável pad é uma constante.

Transferindo variáveis. Variáveis podem ser transferidas para os registros de grupo produzidos. Note que apenas os valores do primeiro caso no grupo são transferidos.

10.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. Variáveis ID definindo os grupos e as variáveis a serem agregadas são especificadas

com os parâmetros. As variáveis ID são automaticamente incluídas no dataset de saída.

Transformando dados. Declarações de Recode podem ser usadas.

Tratamento de dados perdidos. Cada valor de variável agregada é comparado com ambos os códigos de dados perdidos e se for constatado ser um valor de dados perdidos, é automaticamente excluído de qualquer cálculo. Um percentual fornecido pelo usuário, o "ponto de corte" (ver o parâmetro CUTOFF) determina o número de valores de dados perdidos permitido antes do valor de resumo ser produzido como um código de dados perdidos. Portanto, por exemplo, suponha que a média de uma variável agregada dentro de um grupo foi calculada, e o grupo continha 12 registros e 6 deles tinham valores de dados perdidos, i.e. 50%. Se o valor de CUTOFF era de 75%, a média dos 6 valores de dados não-perdidos seria calculada e produzida para aquele grupo. Se o valorde CUTOFF era de 25%, ao contrário, a média não seria calculada e o primeiro código de dados perdidos seria produzido.

10.3 Resultados

Resumo de dados perdidos. (Opcional: ver o parâmetro PRINT). Para cada variável em cada grupo, o número da variável de entrada, o número da variável de saída, o número de registros com dados substantivos (i.e. dados não-perdidos) e o percentual de registros com dados perdidos são impressos.

Resumo do grupos. (Opcional: ver o parâmetro PRINT). O número de registros de entrada em cada grupo.

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C se houver, somente para variáveis usadas na execução.

Dicionário de saída. (Opcional: ver o parâmetro PRINT).

Estatísticas. (Opcional: ver o parâmetro PRINT). Todas as variáveis computadas podem ser impressas para cada registro agregado. O número da variável da variável agregada correspondente e as variáveis ID são também dadas.

10.4 Dataset de Saída

O dataset de saída agrupado é um arquivo Data, descrito em um dicionário do IDAMS. Cada registro contém valores das variáveis ID, variáveis computadas, variáveis transferidas e constantes pad; há um registro produzido para cada grupo.

Seqüência de variáveis e número de variáveis. As variáveis de saída estão na mesma ordem relativa das variáveis de entrada de onde elas são derivadas, a despeito de se a variável de entrada é utilizada como um ID, agregada, ou variável a ser transferida. Portanto, se a primeira variável no entrada é utilizada, as variáveis derivadas disso serão as primeiras variáveis de saída. Cada variável de entrada usada como uma ID ou variável a ser transferida corresponde a uma variável de saída; cada variável agregada corresponde de 1 a 7 variáveis de saída, de acordo com o número de estatísticas requisitadas (essas variáveis são poduzidas em uma ordem relativa: soma, média, variância, desvio-padrão, contagem, mínimo, máximo). As variáveis de saída são sempre renumeradas, começando com um número fornecido no parâmetro VSTART. Constantes pad sempre vêm no final.

Nomes de variáveis. As variáveis de saída possuem o mesmo nome das variáveis de entrada de onde derivam exceto para as variáveis agregadas, os 23° e 24° caracteres dos campos do nome são codificados:

S	=	soma
M	=	média
V	=	variância
D	=	desvio-padrão
CT	=	contagem
MN	=	mínimo
MX	=	máximo.

Constantes pad são batizadas de "Pad variable 1", "Pad variable 2", etc.

Tipo de variável. Variáveis ID e variáveis transferidas são produzidas com o seu tipo de entrada. Variáveis computadas são sempre produzidas como numéricas.

Larguras de campo e número de decimais. Larguras de campo para variáveis agregadas produzidas dependem da estatísitica, da largura do campo de entrada (FW), do número de casas decimais de entrada (ND) e das casas decimais extras requisitadas com o parâmetro DEC. Larguras de campo e número de casas decimais são designadas como mostrado abaixo, onde FW=largura do campo de entrada e ND=número de casas decimais de entrada para variáveis de entrada, e FW=6 e ND=0 para variáveis recodificadas.

Estatística	*Largura de campo*	*Casas decimais*
SUM	FW + 3 *	ND
MEAN	FW + DEC **	ND + DEC ***
VARIANCE	FW + DEC **	ND + DEC ***
SD	FW + DEC **	ND + DEC ***
MIN	FW	ND
MAX	FW	ND
COUNT	4	0

* Se a largura do campo exceder 9, então ele é reduzido a 9.

** Se a largura do campo exceder 9, então o número de casas decimais extras (DEC) é reduzido de acordo.

*** Se o número de casas decimais exceder 9, então DEC é reduzido de acordo.

Códigos de dados perdidos. Códigos de dados perdidos para variáveis ID e variáveis transferidas são retirados do dicionário de entrada. O segundo código de dados perdidos (MD2) para variáveis computadas é sempre um espaço em branco. O valor do primeiro código de dados perdidos (MD1) é alocado como se segue:

Variável de saída	*MD1 produzido*
FW do saída <= 7	9's
FW do saída > 7	-999999
Variável COUNT	9999

Números de referência. Variáveis computadas recebem o número de referência das suas variáveis base.

Registros-C. Registros-C no diconário de entrada são transferidos para o dicionário de saída para variáveis ID e transferidas.

Uma observação para o cálculo de estatísticas. Antes de produzidos, valores computados são arredondados para a largura calculada e número de casas decimais. Se o valor computado excede 999999999 ou é menor do que -99999999, será produzido como 999999999.

10.5 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Variáveis definidoras de grupos (ID) e variáveis a serem transferidas podem ser numéricas ou alfabéticas, apesar de variáveis numéricas serem tratadas como listas de caracteres, i.e. um valor de '044' é diferente de ' 44'. Elas não podem ser variáveis recodificadas. Variáveis a serem agregadas devem ser numéricas e podem ser variáveis recodificadas.

O arquivo é processado de forma serial e registros contíguos com o mesmo valor de variáveis ID são agregados. Portanto, o arquivo de entrada deve ser classificado nas variáveis de ID antes de se usar AGGREG. Note que AGGREG não checa a ordem de classificação do arquivo de entrada.

10.6 Estrutura de Setup

```
$RUN AGGREG

$FILES
     Especificações de arquivo

$RECODE (optional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída
DATAyyyy   dados de saída
PRINT      resultados (default IDAMS.LST)
```

10.7 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V1=10,20,30,50 OR V10=90-300
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para identificar os resultados.

   ```
   Exemplo:  AGGREGATION TEACHER/STUDENT DATA
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  IDVARS=(V1,V2)  STATS=(SUM,VARI)  DEC=3 -
                AGGV=(V5-V10,V50-V75)  PAD1=80
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada. Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores de dados não-numéricos em variáveis agregadas e em variáveis usadas em Recode. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
: O número máximo de casos (depois de filtragem) a ser utilizado do aquivo de entrada.
Default: Todos os casos serão usados.

IDVARS=(lista de variáveis)
: Até 20 números de variáveis para definir os grupos. Variáveis-R não são permitidas.
Não há default.

AGGV=(lista de variáveis)
: Variáveis-R ou -V para serem agregadas.
Não há default.

STATS=(SUM, MEAN, VARIANCE, SD, COUNT, MIN, MAX)
: Parâmetros para selecionar estatísticas requeridas (pelo menos uma das seguintes: SUM, MEAN, VARIANCE, SD deve ser selecionada). Elas são produzidas para cada grupo e para cada variável AGGV.

SUM	Soma.
MEAN	Média.
VARI	Variância.
SD	Desvio-padrão.
COUN	Número de casos válidos.
MIN	Valor mínimo.
MAX	Valor máximo.

SAMPLE/POPULATION

SAMP	Computa a variância e/ou desvio-padrão usando a equação da amostra.
POPU	Use a equação da população.

OUTFILE=**OUT**/yyyy
: Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
Default ddnames: DICTOUT, DATAOUT.

VSTART=**1**/n
: Número da variável para a primeira variável no dataset de saída.

CUTOFF=**100**/n
: O percentual de casos com códigos MD permitidos antes que um código de MD seja produzido. Um valor inteiro.

DEC=**2**/n
: Para variáveis computadas involvendo média, variância ou desvio-padrão: o número de casas decimais em adição aquelas da variável de entrada correspondente (ver Restrição 7).

TRANSVARS=(lista de variáveis)
: Variáveis cujos valores, como designados para o **primeiro** caso de cada grupo, devem ser transferidos para o arquivo de saída. Variáveis-R não são permitidas.

PAD1=constante
PAD2=constante
PAD3=constante
PAD4=constante
PAD5=constante
: Até 5 constantes podem ser adicionadas ao dataset de saída. O número de caracteres dado determina a largura de campo da constantes.

PRINT=(MDTABLES, GROUPS, DATA, CDICT/DICT, **OUTDICT**/OUTCDICT/NOOUTDICT)

MDTA	Imprime uma tabela dando o percentual de dados perdidos encontrados para cada variável agregada em cada grupo.
GROU	Imprime o número de casos por grupo.
DATA	Imprime o valor de cada variável computada em cada registro de grupo.
CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
OUTD	Imprime o dicionário de saída sem registros-C.
OUTC	Imprime o dicionário de saída com registros-C de variáveis ID e de transferência, se houver.
NOOU	Não imprime o dicionário de saída.

10.8 Restrições

1. Número máximo de variáveis a serem agregadas é 400.
2. Número máximo de variáveis ID é 20.
3. Número máximo de caracteres em variáveis ID é 180.
4. Número máximo de variáveis a serem transferidas é 100.
5. Variáveis recodificadas não permitidas como IDVARS ou como TRANSVARS.
6. A mesma variável não pode aparecer em duas listas de variável.

10.9 Exemplo

Produz um dataset contendo um caso agregado para cada valor único de V5 e V7; as variáveis em cada caso devem ser a soma, média e desvio-padrão de 4 variáveis de entrada e 1 variável recodificada, agregada ao longo dos casos formando o grupo (i.e. com os mesmos valores para V5, V7); valores de V10, V11 para o primeiro caso de cada grupo devem ser transferidos para os registros de saída; uma listagem dos valores produzidos para cada caso é requerido; no arquivo de saída, as variáveis devem ser numeradas começando de 1001.

```
$RUN AGGREG
$FILES
PRINT   = AGGR.LST
DICTIN  = IND.DIC                       arquivo Dicionário de entrada
DATAIN  = IND.DAT                       arquivo Dados de entrada
DICTOUT = AGGR.DIC                      arquivo Dicionário de saída
DATAOUT = AGGR.DAT                      arquivo Dados de saída
$RECODE
    R100=COUNT(1,V20-V29)
    NAME R100'WEALTH INDEX'
$SETUP
AGGREGATION OF 4 INPUT VARIABLES AND 1 RECODED VARIABLE
IDVARS=(V5,V7)  AGGV=(V31,V41-V43,R100)  STATS=(SUM, MEAN, SD)   -
   VSTART=1001  PRINT=DATA  TRANS=(V10,V11)
```

Capítulo 11

Construção de um Dataset IDAMS (BUILD)

11.1 Descrição Geral

BUILD pega um arquivo de dados brutos, que pode conter vários registros por caso, juntamente com um dicionário descrevendo as variáveis requeridas e cria um novo arquivo Dados com apenas um registro por caso contendo valores apenas para as variáveis especificadas. Ao mesmo tempo, ele produz um dicionário do IDAMS descrevendo o novo arquivo Dados formatado, em outras palavras um dataset do IDAMS é criado.

Além de reestruturar os dados, BUILD checa também se há valores não-numéricos em variáveis numéricas.

Por que usar BUILD? Qualquer programa do IDAMS pode ser usando sem ter primeiro sido usado BUILD, através da preparação de um dicionário do IDAMS separado. Contudo, BUILD é recomendado como um passo preliminar, dado que:

- propicia a checagem da correta preparação do dicionário,
- assegura que haverá um match perfeito entre o dicionário e os dados,
- assegura que não haverá caracteres não-numéricos inesperados nos dados,
- reduz os dados a um formato compacto de um regitro por caso,
- recodifica todos os espaços vazios de maneira que assumam valores especificados pelo usuário.

Processamento de variáveis numéricas. Quando BUILD processa um campo como contendo um valor numérico, ele checa se o campo possui um número reconhecível ou se possui um espaço em branco. Se um valor diferente disso ocorre, e.g. '3J', '3-', '**2', etc. a posição seqüencial do caso, o número da variável associado com o campo, e o caso de entrada são impressos e uma lista de noves é usada como valor de saída.

As regras de processamento são as seguintes:

- Se um campo contém um número reconhecível, o número é editado em uma forma padrão e, então, processado (ver o capítulo "Dados em IDAMS" para detalhes).
- Se um campo contém somente espaços vazios, ele ou é recodificado para assumir o 1° ou 2° código de dados perdidos, noves ou zeros, ou, se nenhuma recodificação é especificada, é sinalizado como um erro e processado como um campo em branco. A coluna 64 no registros-T pode ser usada para especificar uma regra de recodificação para a variável (ver a seção "Dicionário de Entrada" para detalhes).
- Se um campo contém espaços em branco no final ilegais, e.g. '04 ' em um campo numérico de três dígitos, ou espaços em branco envolvidos, e.g. '0 4', ele é reportado como um erro e o valor é mudado para 9's.
- Se um campo contém um valor positivo ou negativo com caracteres '+' ou '-' erroneamente entrados, e.g. '1-23', ele é reportado como um erro e o valor é mudado para 9's.

- Se um código de dados perdidos para uma variável possui um dígito a mais do que o campo, o campo de saída será um caracter mais comprido do que o de entrada. Essa carcterística pode ser utilizada quando for necessário aumentar o campo de saída sem mudar a largura do campo de entrada; por exemplo, se códigos 0-9 e um espaço em branco fossem definidos para uma única variável de coluna, o campo em branco não poderia ser recodificado em um único valor numérico sem permitir um código de 2-dígitos no saída.

Tabela exibindo exemplos de edição executados por BUILD
e os conteúdos do campo de saída para um campo numérico de entrada de 3-dígitos

Valor do entrada	No. dec.	MD1	Recodificação especif.	Valor do saída	Largura do campo saída	Messagem de erro
032	0	9999	-	0032	4	-
32	0		-	032	3	-
3 2	0		-	999	3	brancos intercalados na var ...
32	0		-	999	3	brancos intercalados na var ...
-03	0		-	-03	3	-
-3	0		-	-03	3	-
- 3	0		-	-03	3	-
3.2	0		-	003	3	-
32	1		-	032	3	-
.32	1		-	003	3	-
3.2	1		-	032	3	-
.32	2		-	032	3	-
.35	1		-	004	3	-
-.3	0		-	-00	3	-
-.3	1		-	-03	3	-
-03	1		-	-03	3	-
	-	8888	1	8888	4	(somente se PRINT=RECODES)
	-		0	000	3	(somente se PRINT=RECODES)
	-		Nenhum		3	brancos na var ...
A32	-		-	999	3	caracteres ruins na var ...
3-2	-		-	999	3	caracteres ruins na var ...

11.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Esse programa não tem condiçêes de selecionar casos de um arquivo de dados de entrada. O filtro padrão não está disponível. Através das descriçêes das variáveis, qualquer subconjunto dos campos dentro de um caso pode ser selecionado para os dados de saída.

Transformando dados. Declaraçêes de Recode não podem ser utilizadas.

Tratamento de dados perdidos. Build não faz distinção entre dados substantivos e valores de dados perdidos. No entanto, campos em branco podem ser substituídos por códigos de dados perdidos, zeros ou noves.

11.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). A coluna "Brule" na listagem do dicionário contém as regras de recodificação para os campos em branco, como especificadas na col. 64 do dicionário de entrada. Note que mensagens de erro para o dicionário estão intercaladas com a listagem do dicionário e não possuem um número de variável. Se o dicionário de entrada não for impresso, os erros podem ser difíceis de serem identificados.

Dicionário de saída. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis (registros-T) são impressos com ou sem registros-C, se houver.

Característica do arquivo de dados de saída. Comprimento do registro do arquivo de dados de saída.

Mensagens de edição de dados. Para cada caso contendo erros, o caso de entrada (até 100 caracteres por linha) e um relatório de erros na ordem do número da variável são impressos.

Mensagens de recodificação de campos em branco. (Opcional: ver o parâmetro PRINT). Para cada caso contendo campos em branco que foram recodificados, uma mensagem sobre esse fato juntamente com os casos de dados de entrada são impressos. Essas mensagens são integradas com as mensagens de edição de dados, se quaisquer erros ocorrerem também no caso.

11.4 Dataset de Saída

BUILD cria um arquivo Dados e um dicionário do IDAMS correspondente, i.e. um dataset do IDAMS. Note que os registros-T sempre definem a localização das variáveis em termos de posição inicial e largura do campo.

O arquivo de dados contém um registro por cada caso. O comprimento de registro é a soma das larguras dos campos de todas as variáveis de saída e é determinado pelo programa BUILD.

Valores de variáveis numéricas. Valores de variáveis numéricas são editadas em uma forma padrão como descrito no parágrafo "Processamento de variáveis numéricas" acima.

Valores de variáveis alfabéticas. Os valores de dados para variáveis alfabéticas não são editados e são os mesmos tanto no entrada quanto no saída.

Largura de variável. Normalmente BUILD determina que a largura de uma variável seja correspondente ao número de caracteres que a variável ocupa nos dados de entrada. Contudo, se um código de dados perdidos possui um dígito significante a mais do que a largura do campo de entrada, a largura do campo de saída será aumentada de um.

Localização da variável. BUILD desiga os campos de saída na ordem numérica das variáveis. Portanto, se as primeiras duas variáveis possuem largura de saída de 5 e 3, localizaçêes 1-5 são designadas para a primeira variável e 6-8 para a segunda variável, etc.

Número de referência e ID do estudo. O número de referência, se não estiver em branco, e a ID do estudo são os mesmos dos seus valores de entrada. Se o número de referência de um registro-T de entrada ou registro-C está em branco, ele é preenchido com o número da variável.

11.5 Dicionário de Entrada

Este descreve aquelas variáveis que devem ser selecionadas para o saída. O formato é descrito no capítulo "Dados em IDAMS" com a coluna 64 no registros-T sendo utilizada para especificar uma regra de recodificação para espaços em branco em uma variável com se segue:

blank	-	nenhuma recodificação para campos em branco,
0	-	recodifique campos em branco com zeros,
1	-	recodifique campos em branco com o 1° código de dados perdidos da variável,
2	-	recodifique campos em branco com o 2° código de dados perdidos da variável,
9	-	recodifique campos em branco com 9's.

Note: A janela Dicionário da Interface do Usuário não dá acesso à coluna 64. Portanto, use o Editor Geral do WinIDAMS (Arquivo/Abrir/Archivo com Editor Geral) ou qualquer outro editor de texto para preencher essa coluna.

11.6 Dados de Entrada

Os dados podem ser qualquer arquivo de registros de comprimento fixo com um ou mais registros por caso, desde que haja o mesmo número de registros por cada caso. O arquivo deve ser classificado por tipo de

registro na ID do caso. Os valores para qualquer variável deve estar localizado nas mesmas colunas do mesmo registro para todos os casos.

Se os dados de entrada contiverem mais de um registro por caso, MERCHECK deve ser sempre usado antes de BUILD para assegurar que cada dado possua o mesmo conjunto de registros para cada caso.

Note que a notação exponencial dos dados não é aceita por BUILD.

11.7 Estrutura de Setup

```
$RUN BUILD

$FILES
     Especificações de arquivo

$SETUP
     1. Título
     2. Parâmetros

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída
DATAyyyy   dados de saída
PRINT      resultados (default IDAMS.LST)
```

11.8 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descriçêes mais aprofundadas das declaraçêes de controle de programa, itens 1-2 abaixo.

1. **Título** (mandatório). Uma linha contendo até 80 caracteres para dar nome aos resultados.

   ```
   Exemplo:  FILE BUILDING STUDY A35
   ```

2. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  MAXERROR=50
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   LRECL=**80**/n
   O comprimento de cada registro de dados de entrada.
   (Usado para checar se as localizaçêes iniciais nos registros-T são válidas).

MAXCASES=n
O número máximo de casos a ser usado do arquivo de entrada.
Default: Todos os casos serão utilizados.

VNUM=**CONTIGUOUS**/NONCONTIGUOUS
CONT Checa se as variáveis estão numeradas em ordem crescente e consecutiva no dicionário de entrada.
NONC Checa apenas se as variáveis estão numeradas em ordem crescente.

MAXERR=**10**/n
O número máximo de casos com erro (espaços em branco não recodificados e valores não-numéricos em variáveis numéricas) antes de BUILD terminar a execução.

OUTFILE=**OUT**/yyyy
Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
Default ddnames: DICTOUT, DATAOUT.

PRINT=(RECODES, CDICT/DICT, **OUTDICT**/OUTCDICT/NOOUTDICT)
RECO Imprime casos de entrada que contém um ou mais campos em branco que foram recodificados.
CDIC Imprime o dicionário de entrada para todas as variáveis com registros-C, se houver.
DICT Imprime o dicionário de entrada sem registros-C.
OUTD Imprime o dicionário de saída sem registros-C.
OUTC Imprime o dicionário de saída com registros-C, se houver.
NOOU Não imprime o dicionário de saída.

11.9 Exemplos

Exemplo 1. Construa um dataset do IDAMS (dicionário e arquivo de dados); registros de dados de entrada possuem um comprimento de registro de 80 com 3 registros por caso; variáveis são numeradas não-contiguamente no dicionário de entrada; variável V2 é a ID completa (colunas 5-10) enquanto variáveis V3 e V4 contêm as duas partes da ID (colunas 5-8, 9-10, respectivamente); campos em branco devem ser substituídos pelo primeiro código de dados perdidos para as variáveis V101, V122, V168, e por zeros para a variável V169; espaços em branco para V123 (idade) devem ser tratados como erros.

```
$RUN BUILD
$FILES
DATAIN  = ABCDATA RECL=80              arquivo Dados de entrada
DICTOUT = ABC.DIC                      arquivo Dicionário de saída
DATAOUT = ABC.DAT                      arquivo Dados de saída
$SETUP
CONSTRUINDO UM CONJUNTO DE DADOS DO IDAMS
VNUM=NONC  MAXERR=200
$DICT
   3   1 169   3
T   1 CÓDIGO DA CIDADE          1 1 1 3                              ID
T   2 ID DO RESPONDENTE           5  10                              ID
T   3 NÚMERO DO DOMICÍLIO         5   8                              ID
T   4 NÚMERO DO RESPONDENTE       9  10                              ID
T 101 POSIÇÃO DO RESP NA FAMIL.  13              0      9      1      QS1
T 122 SEXO                      225              9             1      QS2
T 123 IDADE                      48  49                              QS2
T 168 OCUPAÇÃO                  358  59          99     98      1      QS3
T 169 RENDA                      61  65              99998      0      QS3
```

Exemplo 2. Verificar a presença de caracteres não-numéricos em 4 campos numéricos; o arquivo de dados de entrada possui um registro por caso; registros são identificados por um campo alfabético; as 5 variáveis não numeradas contiguamente; os arquivos de saída normalmente produzidos por BUILD não são requeridos e são definidos como arquivos temporários (extensão TMP), que serão automaticamente deletados pelo IDAMS no final da execução.

```
$RUN BUILD
$FILES
DATAIN  = A:NEWDATA RECL=256            arquivo Dados de entrada
DICTOUT = DIC.TMP                       arquivo temporário Dicionário de saída
DATAOUT = DAT.TMP                       arquivo temporário Dados de saída
$SETUP
CHECANDO E REPORTANDO CARACTERES NÃO-NUMÉRICOS E ESPAÇOS EM BRANCO
VNUM=NONC  LRECL=256  PRINT=NOOU  MAXERR=200
$DICT
   3   1  35   1   1
T   1 NOME DO RESPONDENTE         1  20 1
T  21 IDADE                      21   2
T  22 RENDA                      29   6
T  25 NO. LOCAIS DE TRAB.       129   1
T  35 TÍTULO CIENT.             201   1
```

Capítulo 12

Verificação de Códigos (CHECK)

12.1 Descrição Geral

CHECK verifica se variáveis possuem valores de dados válidos e lista todos os códigos inválidos por ID de caso e número de variável.

Especificação de códigos. Existem duas maneiras nas quais os códigos para variáveis a serem checadas podem ser especificados. Primeiro, as declarações de controle de programa incluem um conjunto de "especificações de código" com os quais se definem as variáveis e seus códigos válidos. Segundo, o usuário pode fornecer uma lista de variáveis cujos códigos válidos devem ser retirados dos registros-C no dicionário. Em qualquer execução de CHECK, o usuário deve aplicar o primeiro método para algumas variáveis e o segundo método para outras. Especificações de código para variáveis em um setup revertem as especificações do dicionário.

Método usado para checagem de valores de dados. Valores de dados para variáveis, tanto numéricas quanto alfabéticas, são checados em relação aos códigos especificados válidos na base de caracter por caracter. Portanto, se uma especificação de códigos válida de 'V2=02,03' é dada, então um valor de ' 2' nos dados será inválido; um espaço em branco no início dos dados não é considerado igual a zero. Se valores de código são especificados com menos dígitos que a largura do campo da variável, assumem-se zeros no início. Portanto, se a especificação 'V2=2,3' é dada onde V2 é uma variável 2-dígitos, valores válidos usados para comparação para os dados serão interpretados como 02, 03, respectivamente. Similarmente, se '-3' e '1' forem dados como códigos válidos para uma variável 3-dígitos, CHECK editará os códigos como '-03' e '001' antes de comparar qualquer valor a eles.

Nota. Se um erro de sintaxe é encontrado em uma especificação de códigos, o resto das especificações é testado mas os dados não são processados.

12.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos do dataset de entrada. O usuário seleciona as variáveis a serem checadas especificando-as em uma "lista de variáveis" e/ou nas "especificações de código".

Transformando dados. Declarações de Recode não podem ser utilizadas.

Tratamento de dados perdidos. CHECK não faz distinção entre dados substantivos e valores de dados perdidos; todos os dados são tratados igualmente.

12.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de dicionário para todas as variáveis são impressos, não apenas para aquelas sendo checados.

Documentação de códigos inválidos. Para cada caso no qual uma variável é encontada com um código inválido, CHECK imprime os valores de ID da variável, as variáveis em erro e seus valores.

12.4 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. CHECK pode checar a validade de dados tanto em variáveis numéricas quanto alfabéticas. Se o dicionário contém registros-C, eles podem ser usados para definir códigos válidos para variáveis.

Valores para variáveis numéricas são assumidos na forma que teriam se fossem editados por BUILD. Essa hipótese implica que não existem espaços em branco no início (eles foram substituídos por zeros), que um sinal negativo, se houver, aparece na posição mais a esquerda, e que casas decimais explícitas não aparecem.

12.5 Estrutura de Setup

```
$RUN CHECK

$FILES
     Especificações de arquivo

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de código (repetido como requerido)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default IDAMS.LST)
```

12.6 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V10=3 AND V20=1-9
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  DATA: THESIS DATA, VERSION 1
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

```
Exemplo:  IDVA=(V1-V4) VARS=(V22-V26,V101-V102)
```

INFILE=**IN**/xxxx
: Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada. Default ddnames: DICTIN, DATAIN.

MAXCASES=n
: O número máximo de casos (depois da filtragem) a ser usado do arquivo de entrada. Default: Todos os casos serão utilizados.

START=**1**/n
: O número seqüencial do primeiro caso a ser checado.

VARS=(lista de variáveis)
: Variáveis cujos códigos válidos devem ser retirados dos registros-C no dicionário.

MAXERR=**100**/n
: Número máximo de casos com códigos inválidos permitido; se esse número é ultrapassado, a execução é terminada.

IDVARS=(lista de variáveis)
: Até 20 variáveis cujos valores devem ser impressos quando um código inválido for encontrado. Isso consistirá, no mínimo, das variáveis que identificam um caso, mas podem ser incluídas outras que forneçam informação adicional para o usuário. As variáveis podem ser alfabéticas ou numéricas. Não há default.

PRINT=CDICT/DICT

CDIC	Imprime o dicionário de entrada para todas as variáveis com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.

4. **Especificações de código** (opcional). Essas especificações definem as variáveis a serem checadas e seus valores de código válidos e inválidos.

```
Exemplos:

V3=1,3,5-9          (Os dados para a variável 3 podem ter códigos 1,3,5-9.
                     Quaisquer outros valores de código são inválidos e
                     serão documentados).

V7,V9,V12-V14= -    (Os dados para as variáveis 7,9 e 12 até 14
  2,50-75,100        podem ter apenas valores de 2,50-75,100).

V50 <> 75           (Os dados para a variável 50 podem ter qualquer código exceto 75).
```

Formato geral

lista de váriáveis = lista de valores de código
ou
lista de váriáveis <> lista de valores de código

Regras de codificação

Cada especificação de código deve iniciar-se em uma nova linha. Para continuar em uma outra linha, quebre depois de uma vírgula e entre com um traço. Podem ser utilizadas quantas linhas de continuação sejam necessárias. Espaços em branco podem ocorrer em qualquer lugar nas especificações.

Lista de variáveis

- Cada número de variável deve ser precedido por um V.
- Variáveis podem ser expressas solitariamente (separadas por uma vírgula), em intervalos (separadas por um traço), ou como uma combinação de ambos (V1, V2, V10-V20).
- As variáveis podem ser definidas em qualquer ordem.
- Todas as variáveis agrupadas em uma expressão devem ter a mesma largura de campo (e.g. para 'V2, V3=10-20' V2 e V3 devem ambas possuir a mesma largura de campo definida no dicionário).
- As variáveis a serem checadas podem ser alfabéticas ou numéricas.

Válido (=) ou inválido (<>)

- Um sinal = indica que valores de código que seguem são os códigos válidos para as variáveis especificadas. Todos os outros códigos serão documentados como erros.
- <> (não igual) indica que os códigos que seguem são inválidos. Todos os casos possuindo esses códigos para as variáveis especificadas serão documentados como erros.

Lista de valores de código

- Códigos podem ser expressos solitariamente (separados por uma vírgula), em intervalos (separados por um traço), ou como uma combinação dos dois.
- Para variáveis numéricas, zeros no início não precisam ser inseridos (e.g. V1=1-10), mas lembre-se de que várias variáveis sendo checadas em relação a códigos em comum devem ter a mesma largura de campo definida no dicionário.
- Para dados com casas decimais, não entre o ponto decimal no valor, mas dê o valor que reflete acuradamente o número assumindo casas decimais implícitas, e.g. o número 2 com uma casa decimal deve ser dado como '20'.
- Para valores alfabéticos, espaços em branco no final não precisam ser entrados; eles são adicionados pelo programa para manter o match com a largura da variável.
- Para definir um espaço em branco ou para especificar um valor contendo espaços em branco intercalados, envolva o valor entre aspas simples (e.g. V10='NEW YORK','PARIS',' ').
- Valores de código podem ser definidos em qualquer ordem.

Notas.

1) Se duas especificações diferentes são dadas para a mesma variável, apenas a última é utilizada.

2) Especificações de código para uma variável reverte o uso de registros de label de código do dicionário para as variáveis fornecidas com o parâmetro VARS.

12.7 Restrições

1. O número máximo de variáveis ID é 20.
2. O número máximo de códigos distintos que pode ser dado em uma especificação de código é 4000. Essa restrição pode ser ultrapassada usando-se intervalos de códigos, pois um intervalo de códigos conta como apenas dois códigos.

12.8 Exemplos

Exemplo 1. Checar a existência de códigos ilegais em variáveis qualitativas e valores além do intervalo em variáveis quantitativas; os únicos códigos válidos para as variáveis V10, V12 e V21 até V25 são 1 a 5 e 9; código 9998 é ilegal para variável V35; códigos 0 e 8 são ilegais para variáveis V41, V44, V46; variáveis V71 a V77 devem ter valores dentro do intervalo 0 a 100, ou 999; casos são identificados pelas variáveis V1, V2 e V4; valores de códigos do dicionário não são utilizados.

```
$RUN CHECK
$FILES
PRINT  = CHECK1.LST
DICTIN = STUDY1.DIC                    arquivo Dicionário de entrada
DATAIN = STUDY1.DAT                    arquivo Dados de entrada
$SETUP
ROTINA PARA PROCURAR CÓDIGOS ILEGAIS E VALORES FORA DO INTERVALO
IDVARS=(V1,V2,V4)
V10,V12,V21-V25=1-5,9
V35<>9998
V41,V44,V46<>0,8
V71-V77=0-100,999
```

Exemplo 2. Checar a validade do código apenas para um subconjunto de casos (quando a variável V21 é igual a 2 ou 3 e a variável V25 é igual a 1); códigos válidos para algumas variáveis são retirados do dicionário de registros-C; em adição, uma especificação de código é dada para a variável V48; casos são identificados pela variável V1.

```
$RUN CHECK
$FILES
DICTIN = STUDY2.DIC                    arquivo Dicionário de entrada
DATAIN = STUDY2.DAT                    arquivo Dados de entrada
PRINT  = CHECK.PRT
$SETUP
INCLUDE V21=2,3 AND V25=1
ROTINA PARA PROCURAR CÓDIGOS ILEGAIS
IDVARS=V1  VARS=(V18-V28,V36-V41)
V48=15-45,99
```

Capítulo 13

Verificação de Consistência (CONCHECK)

13.1 Descrição Geral

CONCHECK usado em conjunto com declarações Recode do IDAMS oferece uma capacidade de checagem de consistência para testar a existência de relações ilegais entre valores de diferentes variáveis. Declarações de condição no setup do CONCHECK são usadas para nomear cada checagem e para indicar quais variáveis devem ser listadas no evento de um erro.

As checagens de consistência são definidas através do Recode testando uma relação lógica e colocando o valor de uma variável de resultado 1 se a relação não é satisfeita, e.g. se V3 não pode logicamente ter o valor de 9 quando V2 tem o valor de 3 então a seguinte declaração de Recode pode ser usada:

IF V2 EQ 3 AND V3 EQ 9 THEN R100=1 ELSE R100=0

Quando uma inconsistência é detectada em um caso, valores de variáveis ID especificadas para cada caso são impressos. Além disso, os valores para um conjunto de variáveis, definido com o parâmetro VARS, são impressos. Esse conjunto é usado para se obter uma visão geral do caso para se detectar a razão da inconsistência de maneira mais fácil e para ter certeza que a correção de uma inconsistência não causará outra. Para cada condição de consistência que falha, um conjunto separado de variáveis, normalmente consistindo de variáveis particulares sendo checadas, podem ser impressas juntamente com o número e nome da condição.

13.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos para checagem. Variáveis que devem ser listadas caso inconsistências ocorram são especificadas com o parâmetro VARS (para o caso) ou CVARS (para uma condição individual).

Transformando dados. Declarações de Recode são usadas para expressar as checagens de consistência requeridas.

Tratamento de dados perdidos. CONCHECK não faz distinção entre dados substantivos e valores de dados perdidos; todos são tratados igualmente.

13.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, apenas para variáveis usadas na execução.

Inconsistências. Para cada caso contendo uma inconsistência, uma linha de identificação é impressa consistindo do número de seqüência do caso e, opcionalmente, os valores das variáveis ID especificadas. Isso é seguido pelos valores das variáveis especificadas com o parâmetro VARS.

Para cada inconsistência individual detectada em um caso, o número e nome da condição correspondente e os valores das variáveis especificadas na declaração de condição são impressos.

Estatísticas de erros. No final da execução, uma tabela de sumário é impressa dando o número de casos processados, o número de casos contendo pelo menos uma inconsistência e, para cada condição de inconsistência, seu número e nome, e o número de casos que falharam o teste.

13.4 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Variáveis numéricas ou alfabéticas podem ser usadas.

13.5 Estrutura de Setup

```
$RUN CONCHECK

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode expressando inconsistências

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Declarações de condição

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default IDAMS.LST)
```

13.6 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, ítens 1-4 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V1=1
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

```
Exemplo:  TESTING FOR INCONSISTENCIES IN NORTH REGION
```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

```
Exemplo:  IDVARS=(V1,V3-V4)  MAXERR=50
```

INFILE=**IN**/xxxx
: Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2
: Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
: O número máximo de casos (depois da filtragem) a ser usado do arquivo de entrada.
Default: Todos os casos serão utilizados.

MAXERR=**999**/n
: O número máximo de inconsistências a serem impressas antes de CONCHECK parar.

IDVARS=(lista de variáveis)
: Até 5 variáveis cujos valores serão listados para identificar casos com inconsistências.
Default: Número seqüencial do caso é impresso.

VARS=(lista de variáveis)
: Variáveis a serem listadas para qualquer caso que possui pelo menos um erro.

FILLCHAR='string'
: Até 8 caracteres usados para separar variáveis ao se listar inconsistências.
Default: 2 espaços.

PRINT=(CDICT/DICT, VNAMES)

CDIC	Imprime o dicionário de entrada para variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
VNAM	Imprime os 6 primeiros caracteres dos nomes das variáveis ao invés dos números das variáveis quando se estiver listando valores de variáveis para casos de inconsistências.

4. **Declarações de condição** (pelo menos uma deve ser dada). Uma declaração de condição é fornecida para cada consistência a ser testada, dando uma referência às declarações de Recode correspondentes, um nome para o teste e as variáveis cujos valores devem ser listados quando o teste falha.

   As regras de codificação são as mesmas dos parâmetros. Cada declaração de condição deve iniciar em uma nova linha.

```
Exemplo:  TEST=R3  CVARS=(V34,V36,V52)  -
            CNAME='AGE, SEX AND PREGNANCY STATUS'
```

TEST=número da variável
: Variável para a qual um valor diferente de zero indica que um teste de consistência falhou.
Não há default.

CVARS=(lista de variáveis)
: Lista de variáveis cujos valores serão listados quando essa inconsistência é encontrada.
Default: Apenas variáveis especificadas com IDVARS e VARS serão listadas.

CNUM=n
Número da condição.
Default: Número de seqüência da condição.

CNAME='string'
Nome para essa condição, até 40 caracteres.
Default: Nenhum nome.

13.7 Restrições

1. Apenas os primeiros 4 caracteres de variáveis alfabéticas são impressos.
2. Nomes de condição não podem ser maiores que 40 caracteres.
3. Número máximo de variáveis ID é 5.
4. Número máximo de variáveis listadas para cada caso em erro (lista VARS) é 20.
5. Número máximo de variáveis listadas para cada condição (lista CVARS) é 20.

13.8 Exemplos

Exemplo 1. Testa a relação entre V6 e V7 e entre V20 e V21; as variáveis de identificação V2 e V3 devem ser impressas para cada caso com um erro juntamente com os valores das variáveis-chave V8-V10; nomes de variáveis devem ser impressos.

```
$RUN CONCHECK
$FILES
PRINT  = CONCH1.LST
DICTIN = MY.DIC                      arquivo Dicionário de entrada
DATAIN = MY.DAT                      arquivo Dados de entrada
$RECODE
  R1=0
  R2=0
  IF V5 INLIST(1-5,8) AND V7 EQ 2 THEN R1=1
  IF V20 LE 3 AND V21 EQ 5 OR V20 EQ 8 AND V21 EQ 7 OR V20 EQ V21 THEN R2=1
$SETUP
TESTANDO 2 INCONSISTÊNCIAS
PRINT=VNAMES  IDVARS=(V2,V3)  VARS=(V8-V10)
TEST=R1  CNAME='1st Inconsistency' CVARS=(V5,V7)
TEST=R2  CNAME='2nd Inconsistency' CVARS=(V20,V21)
```

Exemplo 2. Testar 5 condições na parte 2 de um questionário; testes são numerados começando com 201; todas as variáveis da parte 2 devem ser listadas para cada questionário com erro, juntamente com as variáveis-chave da parte 1 (V5-V10); em adição, variáveis particulares usadas em testes devem ser listadas de novo para cada teste que falha. Note o uso da função SELECT do Recode para inicializar as variáveis de resultado com o valor de 0.

```
$RUN CONCHECK
$FILES
DICTIN = MY.DIC                      arquivo Dicionário de entrada
DATAIN = MY.DAT                      arquivo Dados de entrada
$SETUP
PART 2 DA CHECAGEM DE CONSISTÊNCIA
MAXERR=400  IDVARS=(V1,V3)  VARS=(V5-V10,V200-V231)
TEST=R1  CNUM=201  CVARS=(V203-V205)
TEST=R2  CNUM=202  CVARS=(V203,V210-V212)
TEST=R3  CNUM=203  CVARS=(V214,V215)
TEST=R4  CNUM=204  CVARS=(V222-V226)
TEST=R5  CNUM=205  CVARS=(V229,V230)
$RECODE
      R900=1
A     SELECT (FROM=(R1-R5), BY R900) = 0
      IF R900 LT 5 THEN R900=R900+1 AND GO TO A
      IF V203 IN(1-5,17,20-25) AND V204 EQ 3 OR V205 EQ 'M' THEN R1=1
      IF V203 GT 6 AND MDATA(V210,V211,V212) THEN R2=1
      IF 2*TRUNC(V214/2) EQ V214 OR V215 EQ 0 THEN R3=1
      IF COUNT(1,V222-V226) LT 2 THEN R4=1
      IF MDATA(V229) AND NOT MDATA(V230) THEN R5=1
```

Capítulo 14

Verificação de Intercalação de Registros (MERCHECK)

14.1 Descrição Geral

O programa MERCHECK detecta e corrige erros de fusão (registros perdidos, duplicados ou inválidos) em um arquivo de dados contendo registros múltiplos por caso. Ele produz um arquivo de dados contendo um número igual de registros por caso por meio do preenchimento de registros perdidos e deleção de registros duplicados e inválidos. Apesar de ter sido originalmente escrito para checar dados de imagem de cartão, o comprimento do registro de dados de entrada pode ser qualquer valor até 128. Como qualquer outro programa do IDAMS assume que cada caso no arquivo de dados possue exatamente o mesmo número de registros, o uso de MERCHECK é um primeiro passo essencial para a checagem de todos os arquivos de dados que possuem mais de um registro por caso.

Operação do programa. O usuário fornece um conjunto de descrições de Record definindo os tipos de registros permissíveis. Ao processar os dados, o programa lê em uma área de trabalho todos os dados de entrada contíguos achados que possuem valores de ID do caso idênticos. Esses registros são comparados um a um com os tipos de registro definidos, e um caso de saída é construído. Registros são preenchidos, deletados, reordenados, etc., caso seja necessário. O caso de dados é então transferido para o arquivo de saída, e o programa retorna para ler o conjunto de registros de entrada do próximo caso. Os resultados documentam as correções dos dados de entrada executados pelo programa.

Identificação de casos e de registros. MERCHECK requer que a ID do caso esteja na mesma posição para todos os registros. Campos de ID de casos podem estar localizados em colunas não contíguas e podem ser compostos de qualquer caracteres. Tipos de registros são identificados por um único campo de ID de registro (de 1-5 colunas) que pode ser composto de qualquer caracter exceto espaço em branco. Um esboço de um arquivo de dados com dois tipos de registro segue. Os pontos simbolizam campos em branco ou de dados.

```
...SE23...01..............10......
...SE23...01..............12......
...SE23...02..............10......
...SE23...02..............12......
...SE24...01..............10......
...SE24...01..............12......

 campo    campo           campo de
 de ID do de ID do        ID do
 primeiro segundo         registro
 caso     caso
```

No exemplo, há dois tipos de registro para cada caso, identificados por um 10 ou 12 nas colunas 28, 29. A ID do caso consiste de dois campos não-contíguos, colunas 4-7 e colunas 11-12. Portanto, “SE2301” é uma ID de caso, como são “SE2302” e “SE2401”.

Eliminando registros inválidos. Um registro de dados de entrada contendo uma ID de registro não definido por descrições de Record, conhecido como um registro "extra", é opcionalmente impresso mas nunca transmitido para o arquivo de saída. Além disso, há duas opções para eliminar outros tipos de registros inválidos.

- Registros que não contêm uma constante especificada são rejeitados. (Ver os parâmetros CONSTANT, CLOCATION, e MAXNOCONSTANT).
- O usuário pode fornecer o valor da ID do caso do primeiro caso de dados válidos. Todos os registros contendo um valor de ID de caso menor do que aquele especificado será rejeitado. (Ver o parâmetro BEGINID).

Opções para manusear casos com registros perdidos. O usuário deve selecionar, usando o parâmetro DELETE, uma das três possíveis maneiras de manusear casos incompletos.

1. DELETE=ANYMISSING. Um caso não é produzido se um ou mais dos seus tipos de registro estão faltando.
2. DELETE=ALLMISSING. Um caso não é produzido se nenhuma ID de registro válido é encontrada para uma particular ID de caso.
3. DELETE=NEVER. O programa nunca exclui do arquivo de saída um caso faltando um ou mais registros. Ao contrário, ele constrói um registro para cada tipo de registro faltando e "preenche" seus conteúdos com espaços em branco ou valores fornecidos pelo usuário. Ver o parâmetro PADCH e PAD nas descrições do Record. Preenchimento ocorre em localizações de coluna diferentes dos campos do caso e da ID do registro. O caso apropriado e ID's do registro são sempre inseridos pelo programa.

Opções para manusear casos com registros duplicados. Um registro duplicado é aquele contendo o mesmo ID do caso e ID do registro de outro registro, sem considerar-se os conteúdos dos dois registros. O usuário especifica qual duplicata deve ser mantida se houver mais de um registro de entrada carregando o mesmo ID do caso e ID do registro. Poe exemplo, a opção DUPKEEP=1 faz com que o programa retenha o primeiro registro e descarte quaisquer outros. O caso não é transferido para o arquivo de saída se menos do que n cópias forem encontradas (onde DUPKEEP=n) i.e. para deletar casos com registros repetidos, especifique o valor alto para n. Precaução: Pode acontecer que registros com ID's duplicadas não contenham os mesmos dados. É prerrogativa do usuário determinar a adequação do registro que foi retido.

Opções para manusear registros deletados. Esses registros de dados de entrada que são deletados, i.e. não escritos no arquivo de saída, podem ser salvos em um arquivo separado (ver o parâmetro WRITE).

Seleção dos tipos de registros. MERCHECK permite ao usuário selecionar subconjuntos de tipos de registros de um arquivo de dados de entrada mais amplo. Simplesmente inclua apenas as ID's necessárias nas descrições de Record, e escolha uma opção de impressão de erro apropriada (EXTRAS=n ou PRINT=ERRORS, por exemplo) e um valor realístico de MAXERR. Minimizar o saída impresso para casos com erro é essencial, pois quase todos os casos no arquivo de dados de entrada serão reportados com um erro devido aos registros com ID's de registro inválidas (i.e. aqueles não especificados nas descrições de Record).

Capacidades de recomeçar. O parâmetro BEGINID pode ser usado para recomeçar MERCHECK se uma execução anterior terminou antes de todos os dados de entrada serem processados. O usuário deve determinar o valor da ID do caso para o último caso produzido e fazer BEGINID igual ao valor +1. (Se o fim da execução ocorreu porque o parâmetro MAXERR foi ultrapassado, a última leitura do registro de entrada aparecerá nos resultados, e BEGINID deverá ser ajustada para a ID do caso daquele registro).

Nota. MERCHECK objetiva a checagem de arquivos de dados com registros múltiplos por caso e, portanto, deve haver uma ID de registro em cada registro. MERCHECK poderia teoricamente ser utilizado para eliminar registros duplicados e registros sem uma constante particular para arquivos de dados com um único registro por caso. Isso, contudo, só pode ser feito se cada registro de dados contiver um valor de constante que possa ser tratado como a ID de registro. Essa operação é melhor executada com o programa SUBSET, usando um filtro para excluir registros sem uma constante e a opção DUPLICATE=DELETE para eliminar as duplicatas. (Ver o write-up para SUBSET).

14.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Exceto como definido acima, não disponível para esse programa.

Transformando dados e dados perdidos. Essas opções não se aplicam a MERCHECK.

14.3 Resultados

Casos de erros. O relatório completo com a documentação de cada caso de erro possui três partes: um resumo de erros, os registros não transferidos para o saída (maus registros), e os casos como eles aparecem no arquivo de saída (registros bons). Ver abaixo para maiores detalhes desses componentes. Para dados com um número grande de tipos de registros e com muitos casos com erro, o relatório de casos de erros pode ser custoso e, para alguns trabalhos, completamente desnecessário. O tamanho do relatório necessitado depende de quanto o usuário conhece dos dados, como também da habilidade de corrigir e checar os erros. Por exemplo, se um usuário espera que uma quantidade considerável de preenchimento ocorra, mas virtualmente nenhuma duplicata ou registros inválidos, pode ser suficiente ter apenas um resumo de erros impresso e especificar que casos com erros (se houver) sejam salvos (ver a opção WRITE=BADRECS) e listados mais tarde. Vários controles na quantidade dos resultados são possíveis com os parâmetros PRINT, EXTRAS, DUPS, e PADS.

Casos de erros: resumo de erros. O resumo de erros consiste de uma identificação do caso de erro (contagem do caso ou ID do caso) e qualquer uma das três mensagens sobre os erros que ocorreram. A contagem seqüencial de casos não considera registros ou casos eliminados porque eles aparecem antes da ID do começo ou não possuem a constante requerida. A ID do caso é retirada do(s) campo(s) de ID dos casos como especificado pelo parâmetro IDLOC.

Os três tipos de erros são reportados, ou seja:

1. tipos de registro inválidos,
2. casos com registros perdidos,
3. casos com registros duplicados.

Casos de erros: registros maus. Há os registros inválidos e duplicados, como também registros para casos que foram rejeitados por causa de registros perdidos. Eles são impressos na ordem em que aparecem no arquivo de entrada.

Casos de erros: registros bons. Se um caso é mantido depois de um erro ser encontrado, os registros originais gravados no arquivo de saída, incluindo qualquer registro preenchido, são listados.

Registros ocorrendo antes daquele com BEGINID. Esses são impressos opcionalmente. Ver o parâmetro PRINT=LOWID.

Registros fora da ordem de classificação. Esses são normalmente impressos, apesar de que os resultados não podem ser suprimidos. Ver o parâmetro PRINT=NOSORT.

Registros sem a constante especificada. Qualquer registro que não contém a constante especificada pelo usuário na coluna correta é impresso. Esse relatório pode ser suprimido. Ver o parâmetro PRINT=NOCONSTANT.

Estatísticas de execução. No final do relatório, o número total de registros perdidos e registros duplicados, e o número total de casos que foram lidos, escritos, deletados e que contêm erros são impressos.

14.4 Dados de Saída

Os dados de saída é um arquivo com o mesmo comprimento de registro do arquivo de dados de entrada e um mesmo número de registros por caso. Cada caso contém cada um dos tipos de registro especificado nas descrições do Record.

14.5 Dados de Entrada

A entrada consiste de um arquivo de registros de dados de comprimento fixo normalmente classificado por ID de caso e ID de registro dentro do caso. O comprimento do registro não pode exceder 128.

14.6 Estrutura de Setup

```
$RUN MERCHECK

$FILES
     Especificações de arquivo

$SETUP
     1. Título
     2. Parâmetros
     3. Descrições de registro (repetido como requisitado)

$DATA (condicioinal)
     Dados


Arquivos:
FT02       registros rejeitados (registros de "casos ruins")
           quando WRITE=BADRECS especificado
DATAxxxx   dados de entrada  (omitir se $DATA é usado)
DATAyyyy   dados de saída (casos bons)
PRINT      resultados (default  IDAMS.LST)
```

14.7 Declarações de Controle de Programa

Referir-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-3 abaixo.

1. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  CHECKING THE MERGE OF RECORDS IN STUDY 95 DATA
   ```

2. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  MAXE=25 RECORDS=8 IDLOC=(1,5)
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para o arquivo Dados de entrada.
   Default ddname: DATAIN.

   MAXCASES=n
   O número máximo de casos a ser usado do arquivo de entrada.
   Default: Todos os casos serão utilizados.

   MAXERR=**10**/n
   Número máximo de casos com erros. Quando n + 1 casos de erro ocorrerem, a execução termina. Casos antes do BEGINID, aqueles fora da ordem de classificação, e registros sem a constante não contam como casos de erro. Casos de erro são aqueles com registros inválidos, duplicados, ou perdidos.

OUTFILE=**OUT**/yyyy

Um sufixo ddname de 1-4 caracteres para o arquivo Dados de saída.
Default ddname: DATAOUT.

RECORDS=**2**/n

O número de registros por caso (como definido nas descrições de Record).

IDLOC=(s1,e1, s2,e2, ...)

Colunas iniciais e finais de 1-5 campos de identificação de casos. Pelo menos uma deve ser fornecida. Se houver mais de um campo de ID de caso, então eles devem ser especificados na ordem na qual os dados de entrada são classificados.
Nenhum default.

BEGINID='id do caso'

O menor valor de ID de caso válido na qual o programa começa a processar: de 1 a 40 caracteres entre aspas simples se contiver qualquer caracter não-alfanumérico. Se campos de ID de casos múltiplos são utilizados, o valor deve ser a concatenação das ID's de casos individuais fornecida na ordem de classificação.
Default: Em branco.

NOSORT=**0**/n

O número máximo de casos fora da ordem de classificação tolerado pelo programa. Quando n+1 casos fora da ordem de classificação ocorrerem, a execução termina.

DELETE=**NEVER**/ANYMISSING/ALLMISSING

Especifica sob quais condições, em relação a registros perdidos, um caso deve ser deletado.

- NEVE — Nunca rejeita um caso devido a registros perdidos. Se algum ou todos os registros estão perdidos, o programa irá preencher (com espaços em branco ou valores fornecidos pelo usuário) todos os registros que são perdidos e rejeitar qualquer registro com ID's de registro inválida antes de processar o caso.
- ANYM — Não processa nenhum caso no qual um ou mais registros estão perdidos, i.e. nenhum caso incompleto deve ser processado.
- ALLM — Não processa qualquer caso no qual não haja registros válidos, i.e. quando todos os registros para um caso possuem ID's de registro inválidas.

PADCH=x

Caracter a ser usado em registros preenchidos. Caracteres não-alfanuméricos devem vir entre aspas simples. Ver também descrições de Record para valores de preenchimento mais detalhados.
Default: Em branco.

DUPKEEP=**1**/n

Especifica (para registros de dados duplicados) que a n-ésima duplicata encontrada deve ser mantida. Se menos do que n duplicatas são encontradas, o caso na qual elas ocorrem é deletado (mesmo se DELETE=NEVER é especificado).

WRITE=BADRECS

Cria um arquivo dos registros rejeitados (casos ruins).

CONSTANT=valor

Valor de uma constante. Deve vir entre aspas simples se contiver caracteres não-alfanuméricos. Qualquer registro de dados de entrada sem a constante é rejeitado. A localização da constante deve ser a mesma ao longo de todos os registros de entrada, sem importar o tipo de registro.

CLOCATION=(s, e)

(Fornecida somente se CONSTANT é usada). Localização do campo da constante.

s Coluna inicial do campo de constante em cada registro.

e Coluna final do campo de constante em cada registro.

MAXNOCONSTANT=**0**/n

(Fornecida somente se CONSTANT é usada). Número máximo de registros sem a constante tolerado pelo programa. Quando n + 1 registros sem a constante são encontrados, MERCHECK termina a execução.

PRINT=(**CONSTANT**/NOCONSTANT, **SORT**/NOSORT, **ERRORS**/NOERRORS, LOWID, BADRECS, GOODRECS)

CONS Imprime registros que não possuem a constante especificada.

NOCO Não imprime os registros que não possuem a constante.

SORT Imprime uma notícia de 3-linhas para casos fora de ordem de classificação.

NOSO Não imprime casos fora de ordem de classificação.

LOWI Imprime todos os registros com ID de casos menor do que aquele especificado com BEGINID.

As opções de impressão seguintes referem-se ao relatório de casos com erros (i.e. registros perdidos, inválidos, ou duplicados).

ERRO Imprime o resumo de erros para cada caso com um erro.

NOER Não imprime o resumo de erros para casos com erros.

BADR Imprime registros rejeitados (ruim) para casos com erros.

GOOD Imprime registros mantidos (bom) para casos com erros.

EXTRAS=**0**/n

DUPS=**0**/n

PADS=**0**/n

Se um caso possuir menos do que n registros inválidos (extra/duplicado/preenchido) e nenhum outro erro, nenhum relatório ocorrerá para o caso. Portanto, um caso com apenas 2 registros inválidos e nenhum registro perdido ou duplicado não geraria um relatório caso EXTRAS=3, mas imprimiria de acordo com a especificação do PRINT se tiver também 1 registro perdido.

Default: Todos os casos de erro serão impressos de acordo com a especificação do PRINT.

3. **Descrições de registro** (mandatório: uma para cada tipo de registro para ser selecionado para saída). As regras de codificação são as mesmas das de parâmetro. Cada descrição de registro deve se iniciar em uma nova linha.

```
Exemplo:  RECID=21   RIDLOC=1
          RECID=3    RIDLOC=2   PAD='43599-
          999998889999999881119'
```

RECID=xxxxx

Um código de tipo de registro de 1-5 caracteres que não esteja em branco. Deve vir entre aspas simples se contiver caracteres de letras minúsculas.

Não há default.

RIDLOC=s

Coluna inicial do campo de ID do registro.

Não há default.

PAD='xxx....'

Valores de preenchimento a serem usados ao se preencher um registro desse tipo. A lista de valores deve vir entre aspas simples se contiver caracteres não-alfanuméricos. O primeiro caracter será colocado na coluna 1 do registro preenchido produzido, etc. Para continuar em uma linha subseqüente, entre com um traço. Se o comprimento da string é menor do que o comprimento do registro, então o resto da string é completada na direita com o PADCH especificado na declaração de parâmetro.

Default: PADCH é usado para a string inteira.

Nota: A correta ID de caso e ID de registro são automaticamente inserida dentro de registro preenchido nas posições corretas.

14.8 Restrições

1. Máximo comprimento do registro para dados de entrada é 128.
2. Máximo número de registros de saída por caso é 50.
3. O programa reserva espaço de trabalho para um máximo de 60 registros com valores de ID de caso idênticos. Incluídos na contagem estão registros inválidos, duplicados, e válidos e também registros que são preenchidos pelo programa. MERCHECK termina a execução se mais do que 60 registros com valores de ID de casos idênticos ocorrem na área de trabalho.
4. Máximo comprimento combinado dos campos de ID de casos individuais é de 40 caracteres.
5. Máximo comprimento do campo de ID de registro é de 5 caracteres não-brancos contíguos.
6. Máximo comprimento de uma constante a ser checada é de 12 caracteres.
7. Máximo número de campos de ID de casos é 5.

14.9 Exemplos

Exemplo 1. Checar a fusão de três registros por caso que possuem tipos 1, 2 e 3 respectivamente; registros perdidos são preenchidos: registros 1 e 2 são preenchidos com espaços em branco, registro 3 é preenchido com uma cópia dos valores dados com o parâmetro PAD; casos sem registros válidos (quando todos os registros para um caso possuem tipos de registros inválidos) são escritos no arquivo BAD; casos com até 4 registros duplicados são também escritos no arquivo BAD (se um caso contiver 5 ou mais duplicatas de um tipo particular de registro, então ele é mantido como um bom caso usando a quinta duplicata e eliminando os outros).

```
$RUN MERCHECK
$FILES
PRINT   = MERCH1.LST
FT02    = \DEMO\BAD           arquivo para produzir casos ruins
DATAIN  = \DEMO\DATA1         arquivo Dados de entrada
DATAOUT = \DEMO\DATA2         arquivo Dados de saída (com bons casos apenas)
$SETUP
CHECKING THE MERGE OF DATA
IDLO=(1,3,5,6,10,10)  RECO=3  DELE=ALLM  DUPK=5  WRITE=BADRECS  MAXE=200
RECID=1  RIDLOC=12
RECID=2  RIDLOC=12
RECID=3  RIDLOC=12   PAD='9999999999-
939999999999999999999999999999999999999999999999999999999999999999'
```

Exemplo 2. Cheque os dados, deletando todos os casos com registros perdidos e eliminando casos que não pertencem ao estudo; o arquivo Dados contém dois registros por caso; casos com registros duplicados são mantidos (descartando todos menos o primeiro de um conjunto de registros duplicados); há um tipo de registro TT nas colunas 4 e 5 de um registro e um AB nas colunas 7 e 8 do outro; a ID do estudo, HST, deve aparecer nas colunas 124-126 de cada registro.

```
$RUN MERCHECK
$FILES
FT02    = BAD                 arquivo para produzir casos ruins
DATAIN  = DATA  RECL=126      arquivo Dados de entrada
DATAOUT = GOOD                arquivo Dados de saída (com bons casos apenas)
$SETUP
CHECKING THE MERGE OF DATA
IDLO=(1,3)  RECO=2  WRITE=BADRECS  MAXE=20  -
   CONS=HST  CLOC=(124,126)
RECID=TT  RIDLOC=4
RECID=AB  RIDLOC=7
```

Capítulo 15

Correção de Dados (CORRECT)

15.1 Descrição Geral

CORRECT fornece facilidade de correção para dados em um dataset do IDAMS. Valores de variáveis individuais em casos especificados podem ser corrigidos ou casos inteiros podem ser deletados.

CORRECT é útil para corrigir erros em variáveis individuais para casos específicos como os detectados por exemplo por BUILD, CHECK ou CONCHECK. A preparação de instruções de renovação é fácil. Checagens são realizadas para compatibilidade entre os dados e a correção, e uma boa documentação é impressa descrevendo todas as correções realizadas.

Operação do programa. CORRECT inicialmente lê o dicionário e armazena a informação sobre as variáveis em um dataset. Cada instrução de correção de dados é então processada. Depois que a instrução é lida, CORRECT lê o arquivo de dados copiando casos até o caso identificado na instrução ser encontrado. CORRECT executa a instrução, listando o caso, ou revisando valores para variáveis selecionadas e produzindo o caso, ou deletando o caso da saída quando apropriado. Quando todas as instruções são exauridas, os casos de dados remanescentes (se houver) são copiados para a saída, e a execução termina normalmente. Se erros na ordem de classificação das instruções de correção ou casos de dados ocorrem e também se há erros de sintaxe nas intruções de correção, CORRECT documenta a situação nos resultados e continua com a próxima instrução.

Correção de variáveis. O usuário especifica a identificação de caso seguido pelos números de variáveis a serem corrigidos juntamente com os seus novos valores. Ambas varáveis numéricas (com valor inteiro ou decimal) e alfabéticas podem ser corrigidas.

Corrigindo variáveis de ID de casos. Se um campo de ID deve ser corrigido, normalmente a ordem de classificação será afetada e o parâmetro CKSORT=NO deve, portanto, ser especificado. Se a variável de ID contém caracteres não-numéricos errados, então coloque os seus valores entre aspas simples na instrução de correção.

Deleção de casos. O usuário pode deletar um caso do arquivo de dados ao especificar informação de identificação do caso e a palavra “DELETE”.

Listagem de casos. O usuário pode escolher ter um caso particular de dados listado ao especificar informação de identificação do caso e a palavra “LIST”.

15.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Pode-se selecionar um subconjunto de casos a ser processado e produzido incluindo-se um filtro padrão. Seleção de variáveis é inapropriada.

Transformando dados. Declarações de Recode não podem ser utilizados.

Tratamento de dados perdidos. CORRECT não faz distinção entre valores de dados substantivos e dados perdidos; o conceito não se aplica à operação do programa.

15.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de dicionário para todas as variáveis são impressos, não apenas para aqueles sendo corrigidos.

Listagem das intruções de correção. Instruções de correção são sempre listadas. Com cada correção o programa também lista opcionalmente: (1) registros de dados de entrada, (2) registros deletados, ou (3) registros corrigidos (ver o parâmetro PRINT).

15.4 Dataset de Saída

Uma cópia do dicionário é sempre produzida. Se não for requerida, a definição do arquivo DICTOUT pode ser omitida. Os dados são sempre copiados para a saída, mesmo que não haja nenhuma correção ou deleção.

15.5 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Normalmente, CORRECT espera que os casos de dados sejam classificados em ordem ascendente de valores de variáveis ID de caso. O usuário pode, contudo, indicar (via o parâmetro CKSORT) que os casos não estão em ordem ascendente. Essa opção deve ser usada com cuidado: a ordem das instruções de correção deve combinar exatamente com a ordem dos dados no arquivo.

15.6 Estrutura de Setup

```
$RUN CORRECT

$FILES
     Especificações de arquivo

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Instruções de correção (repetida como requerido)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Files:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída
DATAyyyy   dados de saída
PRINT      resultados (default  IDAMS.LST)
```

15.7 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descrições aprofundadas das declarações de controle do programa, ítens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V1=10,20,30 AND V12=1,3,7
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  CORRECTION OF ALPHA CODES IN 1968 ELECTION
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  PRINT=CORRECTIONS, IDVARS=V4
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   MAXCASES=n
   O número máximo de casos (depois da filtragem) a ser utilizado do arquivo de entrada. Se MAXC=0, todas as instruções de correção serão checadas em relação a erros de sintaxe, mas nenhum dado será processado.
   Default: Todos os casos serão utilizados.

   IDVARS=(lista de variáveis)
   Até 5 números de variáveis para os campos de identificação de casos. Se mais de um campo de ID de caso for especificado, os números das variáveis devem ser dados da maior para a menor ordem de campo de classificação.
   Não há default.

   CKSORT=**YES**/NO
   Indica se os casos de dados terão seus campos de ID de caso checados em relação ao ordenamento seqüencial ascendente. A execução termina se um caso fora da ordem é detectado.

   OUTFILE=**OUT**/yyyy
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
   Default ddnames: DICTOUT, DATAOUT.

   PRINT=(DELETIONS, CORRECTIONS, CDICT/DICT)

DELE	Lista aqueles casos para os quais a opção delete é especificada em instruções de correção.
CORR	Lista casos corrigidos.
CDIC	Imprime o dicionário de entrada para todas as variáveis com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.

4. **Instruções de correção.** Essas declarações indicam qual dentre as opções de listagem, deleção, ou correção devem ser aplicadas e para quais casos.

   ```
   Exemplos:

   ID=1026,V5=9,-          (Para o caso com ID "1026" mude o
       V6=22                valor de V5 para 9 e o valor de V6 para 22)
   ID='JOHN DOE',DELETE    (Delete o caso com ID "JOHN DOE" da saída)
   ID=091,3,LIST           (Liste o caso com ID "091", "3")
   ID=023,16,V8='DON_T',-  (Mude V8 para DON'T e V9 para TEACH,RES)
       V9='TEACH|RES'
   ```

Regras para codificação

Cada instrução de correção deve começar em uma nova linha. Para continuar em uma nova linha, quebre depois da vírgula no final de uma correção de variável completa e entre com um traço. Quantas linhas de continuação forem necessárias poderão ser utilizadas. Espaços em branco podem ocorrer em qualquer lugar nas instruções.

As instruções de correção devem ser ordenadas, na mesma seqüência relativa exata, pelos valores de ID do caso, como feito nos casos de dados.

Valores de ID de caso

- O caso a ser corrigido é identificado pela palavra-chave "ID=" seguida pelos valores das variáveis de ID.
- A lista de valores na instrução não está entre parênteses.
- Cada valor, incluindo o último, deve ser seguido por uma vírgula, e a ordem dos valores deve corresponder a ordem das variáveis na lista de variáveis de ID especificada com o parâmetro IDVARS.
- O número de dígitos ou caracteres em um valor deve ser igual à largura da variável como definida no dicionário, i.e. zeros no início talvez precisem ser incluídos.
- Valores contendo caracteres não-numéricos devem vir entre aspas simples, e.g. ID=9,'PAM'.

Tipo de instrução

A identificação de caso é seguida ou pela palavra "LIST", pela palavra "DELETE", ou por uma string de correções de variável.

Correções de variável

- Uma correção de variável consiste de um número de variável precedido por um "V" e seguido por um "=" e o valor correto, e.g. V3=4.
- Correções de variáveis para variáveis diferentes do mesmo caso são separadas por vírgulas.
- Valores de correção para variáveis numéricas podem ser especificados sem zeros iniciais.
- Se a variável inclui casa decimais, o ponto decimal pode ser inserido, mas não é escrito no arquivo de saída. Os dígitos são alinhados de acordo com o número de casas decimais indicado no dicionário e dígitos decimais em excesso são arredondados.
- Se o valor contém caracteres não-numéricos, ele deve vir entre aspas simples. Uma vírgula inclusa deve ser representada como uma barra vertical e uma aspa simples inclusa deve ser representada como um underscore; o programa converterá a barra vertical e o underscore na vírgula e aspas respectivamente, e.g. v8='Don_t'.
- Valores de correção para variáveis alfabéticas devem casar com a largura da variável. Se o valor de correção contém espaços em branco ou caracteres minúsculos, ele deve vir entre aspas simples.

15.8 Restrição

O número máximo de variáveis de ID de caso é 5.

15.9 Exemplo

Correção do arquivo de dados; tanto variáveis numéricas quanto alfabéticas devem ser corrigidas, e dois casos devem ser deletados; casos são identificados pelas variáveis V1, V2 e V5; o dicionário não é mudado, e, portanto, um dicionário de saída não é necessário.

```
$RUN CORRECT
$FILES
PRINT   = CORRECT1.LST
DICTIN  = DATA1.DIC               arquivo Dicionário de entrada
DATAIN  = DATA1.DAT               arquivo Dados de entradaa
DICTOUT = DATA2.DIC               arquivo Dicionário de saída (mesmo de entrada)
DATAOUT = DATA2.DAT               arquivo Dados de saída (corrigido)
$SETUP
CORRECTING A DATA FILE
IDVARS=(V1,V2,V5)
ID=311,01,21,V12='JOHN MILLER'
ID=311,05,41,DELETE
ID=557,11,32,V58=199,V76=2,V90=155
ID=559,11,35,V12='AGATA CHRISTI',V13='F'
ID=657,31,11,V58=100,V77=4,V90=105,V36=999999,V37=999999,V38=999999, -
   V41=98,V44=99
ID=711,15,11,DELETE
```

Capítulo 16

Importação/Exportação de Dados (IMPEX)

16.1 Descrição Geral

O programa IMPEX executa importação/exportação de dados em formato livre e DIF, e importação/exportação de matrizes em formato livre. Em estilo de formato livre, os campos podem ser separados por espaços, tabuladores, vírgula, ponto-e-vírgula ou qualquer caracter definido pelo usuário. Pontos decimais ou vírgulas podem ser usados para a notação decimal. Arquivos de dados importados/exportados podem conter números de variáveis e/ou nomes de variáveis como headings de colunas. Arquivos de matriz importados/exportados podem conter números de variáveis/valores de códigos e/ou nomes de variáveis/labels de códigos como headings de coluna/linha.

Importação de dados. O programa cria um novo dataset do IDAMS de um arquivo ASCII de dados já existente em formato livre ou DIF (formato para intercâmbio de dados desenvolvido pela Software Arts Products Corp.,) e de um dicionário do IDAMS. O dicionário de entrada define como os campos dos dados de arquivo de entrada devem ser transferidos para o dataset IDAMS de saída.

Exportação de dados. O programa cria um novo arquivo de dados ASCII contendo variáveis de um dataset existente do IDAMS e novas variáveis definidas pelas declarações de Recode do IDAMS. O arquivo exportado pode ser em formato livre ou DIF.

Importação de matriz. O programa cria um arquivo Matriz do IDAMS de um arquivo ASCII de formato livre contendo um triangular inferior de uma matriz quadrada ou uma matriz retangular.

Exportação de matriz. O programa cria um arquivo ASCII contendo todas as matrizes armazenadas em um arquivo Matriz do IDAMS. Para a exportação de matrizes, apenas o formato livre está disponível.

16.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos do dados de entrada quando a exportação de dados é requisitada. Da mesma maneira, na exportação de dados, variáveis são selecionadas através do parâmetro OUTVARS.

Transformando dados. Declarações de Recode podem ser usadas na exportação de dados.

Tratamento de dados perdidos. Nenhuma checagem de dados perdidos é feita nos valores de dados, exceto através do uso de declarações de Recode, na exportação de dados. Na importação de dados, campos vazios (campos vazios entre delimitadores consecutivos) são substituídos pelo primeiro código de dados perdidos ou por um campo de 9's se o primeiro código de dados perdidos não está definido.

16.3 Resultados

Importação de dados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, para todas as variáveis inclusas no dicionário.

Labels e códigos de colunas de entrada. (Opcional: ver os parâmetros PRINT e EXPORT/IMPORT). Labels de coluna e códigos de coluna são impressos (sem formatação) ao serem lidos do arquivo de entrada.

Dados de entrada. (Opcional: ver o parâmetro PRINT). Linhas de dados de entrada não formatadas são impressas para todos os casos exatamente como são lidos do arquivo de dados de entrada.

Dicionário de saída. (Opcional: ver o parâmetro PRINT).

Dados de saída. (Opcional: ver o parâmetro PRINT). Valores para todos os casos e todas as variáveis são dados, 10 valores por linha, na mesma ordem das linhas de dados de entrada.

Exportação de dados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, apenas para variáveis usadas na execução.

Dados de saída. (Opcional: ver o parâmetro PRINT). Valores para todos os casos para cada variável-R ou -V são dados, 10 valores por linha. Para variáveis alfabéticas, apenas os 10 primeiros caracteres são impressos.

Importação de matrizes

Matriz de entrada. (Opcional: ver o parâmetro PRINT). Uma matriz contida no arquivo ASCII de entrada é impressa com ou sem labels de coluna e códigos de coluna.

Exportação de matrizes

Matrizes de entrada. (Opcional: ver o parâmetro PRINT). Matrizes contidas no arquivo de matriz do IDAMS de entrada são impressas com ou sem registros de descrição de variável ou registros de código-label.

16.4 Arquivos de Saída

Importação

A saída é um dataset do IDAMS ou uma matriz do IDAMS dependendo se a importação de dados ou matrizes é requisitada.

No caso de um dataset do IDAMS, valores de variáveis numéricas são editados de acordo com as regras do IDAMS (ver o capítulo “Dados em IDAMS”).

Campos numéricos vazios (i.e. strings vazias entre caracteres delimitadores) em um arquivo de entrada em formato livre são substituídos pelo primeiro código de dados perdidos correspondente ou por 9’s se o primeiro código de dados perdidos não está definido.

Exportação

A saída é um arquivo ASCII, cujo conteúdo varia de acordo com os requerimentos de exportação.

Dados em formato DIF. Esse é um arquivo com seções ‘Header” e “Data”. Vetores correspondem a variáveis do IDAMS, e “TUPLES” a casos. Além do itens requeridos pelo header, LABEL (um item opcional padrão) é usado para exportar nomes de variáveis. Na seção Data, o indicador Value “V” é sempre usado para valores numéricos. Um ponto decimal ou vírgula é usada em notação decimal se o número de decimais definido no dicionário for maior que zero.

Dados em formato livre. Este é um arquivo cujos valores das variáveis são separados por um delimitador (ver os parâmetros WITH e DELCHAR) e casos são separados adicionalmente pelo "carriage return" mais caracteres de alimentação de linha. Para valores de variáveis numéricas, um ponto decimal ou vírgula (ver o parâmetro DECIMALS) é incluído se o número de decimais definido no dicionário for maior que zero. Valores de variáveis alfabéticas vêm entre aspas simples ou aspas, ou não inclusos entre quaisquer caracteres especiais (ver o parâmetro STRINGS).

Matriz em formato livre. O formato das matrizes produzidas por IMPEX é o mesmo formato requerido para matrizes importadas (ver "Importação de matrizes" na seção "Arquivos de Entrada" abaixo). A única diferença é que caracteres delimitadores adicionais são inseridos para assegurar o correto posicionamento das labels de coluna e de linha em um software de planilha.

16.5 Arquivos de Entrada

Importação de dados

Para importação de dados, a entrada é:

- um arquivo ASCII contendo um arranjo de dados em formato livre cujos campos são separados por um delimitador, e um dicionário que define como transferir dados para um dataset do IDAMS (todos os campos devem ser descritos no dicionário de entrada);
- um arquivo de dados em formato DIF, e, também, um dicionário de IDAMS.

Os arquivos de entrada podem também conter informação do dicionário. Para arquivos em formato livre, isso significa que labels de coluna e códigos de coluna (que correspondem a nomes de variáveis e número de variáveis) são fornecidos pelo arranjo de dados como as primeiras linhas do arranjo. Ambos, labels e códigos, são opcionais. Se fornecidas, labels de coluna substituem nomes de variáveis do dicionário de entrada, e elas são inseridas no dicionário de saída. Eles podem vir entre caracteres especiais (ver o parâmetro STRINGS). Códigos de coluna são usados apenas para executar uma checagem em relação a números de variáveis do dicionário de entrada. Para arquivos em formato DIF, labels de coluna aparecem como itens LABEL na seção Header. Códigos de colunas podem estar presentes como a primeira linha no arranjo de dados.

Importação de matriz

A entrada é sempre um arquivo ASCII em formato livre onde os valores numéricas/strings de caracteres são separadas por um delimitador. Campos vazios (i.e. strings vazias entre delimatores de caracteres) são pulados. Cada arquivo pode conter apenas uma matriz para importação.

O arquivo de matriz de entrada pode opcionalmente oferecer informação consistindo de uma série de strings para nomear colunas/linhas da matriz e dos códigos correspondentes. Se fornecidos, eles devem seguir a sintaxe dada abaixo (que é diferente para matrizes retangulares e quadradas).

Matriz retangular

Esse é um arquivo ASCII contendo um arranjo rectangular de valores em formato livre; informação do dicionário pode ser incluída opcionalmente.

Exemplo.

```
Salário médio; Grupo de idade; Sexo;
Masculino; Feminino;
1;2;
20 - 30;1;600;530;
31 - 40;2;650;564;
41 - 60;3;723;618;
```

Formato.

1. As primeiras três strings contêm, respectivamente: (1) uma descrição do conteúdo da matriz, (2) o título da linha ("nome da variável de linha"), e (3) o título da coluna ("nome da variável de coluna"). (Opcional).
2. Labels de coluna. (Opcional: uma label por coluna do arranjo de valores).
3. Códigos de coluna. (Opcional: um código por coluna do arranjo de valores).
4. Um arranjo de valores. (Isso pode conter opcionalmente uma label de linha e/ou código antes de cada linha de valores).

Nota. Se labels de linha ou coluna e/ou códigos não estão presentes, eles são automaticamente gerados para a matriz IDAMS de saída (labels como R-#0001, R-#0002, ... C-#0001, C-#0002, ... e códigos de 1 até o número de linhas ou colunas respectivamente).

Matriz quadrada

Esse é um arquivo ASCII contendo a porção triangular inferior de uma matriz (apenas elementos fora da diagonal), e opcionalmente vetores de médias e desvios-padrões seguindo a matriz, em formato livre.

Exemplo.

```
;;Paris;London;Brussels;Madrid; ...
;;1;2;3;4; ...
Paris;1;
London;2;0.55;
Brussels;3;0.45;0.35;
Madrid;4;1.45;2.35;1.15;
. . .
```

Formato.

1. Labels de coluna ("nomes de variáveis"). (Opcional: tantas labels quantas colunas/linhas no arranjo de valores).
2. Códigos de colunas ("números de variáveis"). (Opcional: tantos códigos quantas colunas/linhas no arranjo de valores).
3. Um arranjo de valores. (Pode opcionalmente conter uma label de linha e/ou código antes de cada linha de valores).
4. Um vetor de médias. (Opcional).
5. Um vetor de desvios-padrões. (Opcional).

Nota. Se labels e/ou códigos não estão presentes, eles são automaticamente gerados para a matriz do IDAMS de saída (labels como V-#0001, V-#0002, ... e códigos de 1 até o número de colunas/linhas).

Exportação de dados e matrizes

Dependendo se dados ou matrizes serão exportados, a entrada é um arquivo de dados descrito por um dicionário do IDAMS (ambas, variáveis numéricas e alfabéticas, podem ser usadas) ou um arquivo do IDAMS de matriz(es) quadrada(s) ou retangular(es).

16.6 Estrutura de Setup

```
$RUN IMPEX

$FILES
     Especificações de arquivo

$RECODE (opcional com exportação de dados; não disponível em outros casos)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada para exportação/importação de dados
           (omitir se $DICT é usado)
DATAxxxx   dados/matriz de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída para importação de dados
DATAyyyy   dados/matriz de saída
PRINT      resultados (default  IDAMS.LST)
```

16.7 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, ítens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução se a exportação de dados for especificada.

   ```
   Exemplo:  EXCLUDE V19=2-3
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  EXPORTING SOCIAL DEVELOPMENT INDICATORS
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  EXPORT=(DATA,NAMES)  FORMAT=DELIMITED WITH=SPACE
   ```

   IMPORT=(**DATA**/MATRIX, NAMES, CODES)

DATA	Importação de dados é requisitada.
MATR	Importação de matriz é requisitada.
NAME	Nomes de variáveis são incluídos no arquivo Dados para importação. Nomes de variáveis/labels de códigos são incluídas no arquivo Matriz para importação.
CODE	Números de variáveis são incluídos no arquivo Dados para importação. Números de variáveis/valores de códigos são incluídas no arquivo Matriz para importação.

EXPORT=(**DATA**/MATRIX, NAMES, CODES)

DATA Exportação de dados é requisitada.

MATR Exportação de matriz é requisitada.

NAME Nomes de variáveis devem ser exportados no arquivo Dados de saída. Nomes de variáveis/labels de códigos devem ser exportados no arquivo Matriz de saída.

CODE Números de variáveis devem ser exportados no arquivo Dados de saída. Números de variáveis/valores de códigos devem ser exportados no arquivo Matriz de saída.

Nota. Não há defaults. Ou IMPORT ou EXPORT (mas não ambos) deve ser especificado.

INFILE=**IN**/xxxx

Um sufixo ddname de 1-4 caracteres para os arquivos de entrada:
Arquivo Dados ou Matriz para importação (default ddname: DATAIN),
Arquivos Dicionário e Dados para exportar dados (default ddnames: DICTIN, DATAIN),
Arquivo Matriz IDAMS para exportar (default ddname: DATAIN).

BADDATA=**STOP**/SKIP/MD1/MD2

Tratamento de valores não-numéricos importados ou exportados e valores de saída com "largura de campo insuficiente". Ver capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n

Aplicável apenas se importação/exportação de dados for especificada.
O número máximo de casos (depois da filtragem) a ser usado do arquivo de dados de entrada.
Default: Todos os casos são utilizados.

MAXERR=**0**/n

O número máximo de erros de "largura de campo insuficiente" permitido antes da execução parar. Esses erros ocorrem quando o valor de uma variável é muito grande para caber no campo apontado, e.g. um valor de 250 quando a largura do campo de 2 foi especificada.

OUTFILE=**OUT**/yyyy

Um sufixo ddname de 1-4 caracteres para os arquivos de saída:
Arquivos de Dicionário e Dados obtidos pela importação (default ddnames: DICTOUT, DATAOUT),
Arquivo Matrix do IDAMS obtido por importação (default ddname: DATAOUT),
Arquivo Dados ou Matrix exportado (default ddname: DATAOUT).

OUTVARS=(lista de variáveis)

Aplicável somente se a exportação de dados é especificada.
Variáveis-R e -V que devem ser exportadas. A ordem das variáveis na lista não é significante, pois elas são produzidas em ordem numérica ascendente. Todas os números das variáveis-R e -V devem ser únicos.
Não há default.

MATSIZE=(n,m)

Aplicável apenas se importação de matriz não é especificada.
Número de linhas e colunas da matriz a ser importada. O programa assume uma matriz retangular se ambas são especificadas e uma matriz quadrada simétrica se um deles é omitido.

n Número de linhas.

m Número de colunas.

Não há default.

FORMAT=**DELIMITED**/DIF

Especifica o formato dos dados/matriz de entrada para importação, ou o formato dos dados/matriz de saída para exportação.

DELI Esperam-se dados/matrizes em formato livre, onde os campos são separados por um delimitador (ver abaixo).

DIF Dados é esperado ser em formato DIF.

Nota: o formato DIF está disponível apenas para exportação ou importação de dados.

WITH=**SPACE**/TABULATOR/COMMA/SEMICOLON/USER

(Condicional: ver FORMAT=DELIMITED).

Especifica o caracter delimitador para separar campos em arquivo de formato livre.

SPAC Caracter em branco (código ASCII: 32).

TABU Caracter Tabulador (código ASCII: 9).

COMM Vírgula "," (código ASCII: 44).

SEMI Ponto-e-vírgula ";" (código ASCII: 59).

USER Caracter especificado pelo usuário (ver o parâmetro DELCHAR abaixo).

Nota: Ao importar/exportar arquivos DIF, COMMA é sempre usado com caracter delimitador, independentemente do que é selecionado.

DELCHAR='x'

(Condicional: ver o parâmetro WITH=USER acima).

Define o caracter a ser usado para separar campos em arquivos em formato livre.

Default: Em branco.

DECIMALS=**POINT**/COMMA

Define o caracter usado como notação decimal.

POIN Ponto "." (código de ASCII: 46).

COMM Vírgula "," (código de ASCII: 44).

STRINGS=**PRIME**/QUOTE/NONE

Define o caracter usado para "envolver" as strings de caracteres.

PRIM Aspas simples.

QUOT Aspas.

NONE Nenhum caracter especial é usado.

Nota: ao importar/exportar arquivos DIF, QUOTE é sempre usado, independente do que é selecionado.

NDEC=**2**/n

Número de casas decimais a serem retidas na exportação.

PRINT=(**DICT**/CDICT/NODICT, DATA)

DICT Imprime o dicionário sem registros-C.

CDIC Imprime o dicionário com registros-C, se houver.

DATA Imprime os valores de dados.

Nota:

(a) Opções de impressão de dicionário controlam tanto a impressão do dicionário de entrada quanto o de saída.

(b) Opções de impressão de dados controlam a impressão de dados de saída se um arquivo de dados é exportado, e controlam tanto o entrada quanto o saída se importação de dados é requerida (o entrada nunca é impresso se um arquivo em formato DIF é importado).

(c) Para matrizes, a matriz de entrada é impressa toda vez que impressão de dados é especificada.

16.8 Restrições

1. O número máximo de variáveis-R que podem ser exportadas é 250.
2. O número máximo de variáveis que podem ser usadas em uma execução (incluindo variáveis usadas apenas em declarações de Recode) é 500.
3. O número máximo de linhas da matriz é 100.
4. O número máximo de colunas da matriz é 100.
5. O número máximo de células da matriz é 1000.

16.9 Exemplos

Exemplo 1. Variáveis selecionadas do conjunto de dados de entrada são transferidas para o arquivo de saída juntamente com duas novas variáveis; dados são produzidos em formato livre com valores separados por um ponto-e-vírgula; vírgulas serão utilizadas em notação decimal enquanto valores de variáveis alfabéticas virão entre aspas; nomes e números de variáveis serão incluídos no arquivo de dados de saída.

```
$RUN IMPEX
$FILES
PRINT   = EXPDAT.LST
DICTIN  = OLD.DIC                       arquivo Dicionário de entrada
DATAIN  = OLD.DAT                       arquivo Dados de entrada
DATAOUT = EXPORTED.DAT                  arquivo Dados exportado
$SETUP
EXPORTING IDAMS FIXED FORMAT DATA TO FREE FORMAT DATA
EXPORT=(DATA,NAMES,CODES)  BADD=MD1  MAXERR=20  -
   OUTVARS=(V1-V20,V33,V45-V50,R105,R122)  -
   FORMAT=DELIM  WITH=SEMI  DECIM=COMMA  STRINGS=QUOTE
$RECODE
   R105=BRAC(V5,15-25=1,<36=2,<46=3,<56=4,<66=5,<90=6,ELSE=9)
   MDCODES R105(9)
   NAME R105'GROUPS OF AGE'
   IF MDATA(V22) THEN R122=99.9 ELSE R122=V22/3
   MDCODES R122(99.9)
   NAME R122'NO ARTICLES PER YEAR'
```

Exemplo 2. Dados em formato DIF são importados para o IDAMS; labels e códigos de coluna são incluídos no arquivo de dados de entrada, e vírgulas são utilizadas na notação decimal.

```
$RUN IMPEX
$FILES
PRINT   = IMPDAT.LST
DICTIN  = IDA.DIC             Arquivo Dicionário descrevendo os dados a serem importados
DATAIN  = IMPORTED.DAT        Arquivo Dados a ser importado
DICTOUT = IDAFORM.DIC         Arquivo Dicionário de saída
DATAOUT = IDAFORM.DAT         Arquivo Dados de saída
$SETUP
IMPORTING DIF FORMAT DATA TO IDAMS FIXED FORMAT DATA
IMPORT=(DATA,NAMES,CODES)  BADD=MD1  MAXERR=20  -
   FORMAT=DIF  DECIM=COMMA
```

Exemplo 3. Um conjunto de matrizes retangulares criado pelo programa TABLES é exportado; valores serão separados por um ponto-e-vírgula e vírgulas serão utilizadas na notação decimal; labels e códigos de coluna e linha serão incluídos no arquivo matriz de entrada; matrizes de entrada são impressas.

```
$RUN IMPEX
$FILES
PRINT   = EXPMAT.LST
DATAIN  = TABLES.MAT                    arquivos com matrizes retangulares
DATAOUT = EXPORTED.MAT                  arquivos com matrizes exportadas
$SETUP
EXPORTING IDAMS RECTANGULAR FIXED FORMAT MATRICES TO FREE FORMAT MATRICES
EXPORT=(MATRIX,NAMES,CODES)  PRINT=DATA  -
   FORMAT=DELIM  WITH=SEMI  DECIM=COMMA  STRINGS=QUOTE
```

Exemplo 4. Importando uma matriz quadrada contendo medidas de distância de 10 objetos numerados de 1 a 10; apenas valores inteiros são incluídos e são separados por um sinal % ; códigos de colunas e linhas, tanto quanto vetores de médias e desvios-padrões são incluídos no arquivo de matriz.

```
$RUN IMPEX
$FILES
PRINT   = IMPMAT.LST
DATAOUT = IMPORTED.MAT                  arquivo com a matriz importada
$SETUP
IMPORTING A FREE FORMAT MATRIX TO THE IDAMS SQUARE FIXED FORMAT MATRIX
IMPORT=(MATRIX,CODES)  MATSIZE=10  -
   FORMAT=DELIM  WITH=USER  DELCH='%'
$DATA
$PRINT
%   1%   2%   3%   4%   5%   6%   7%   8%   9%  10%
   1%
   2%38%
   3%72%25%
   4%24%53%17%
   5%64%26%76%18%
   6%48%25%63%15%61%
   7%12%50%7%42%8%8%
   8%19%7%13%4%14%1%15%
   9%29%37%34%21%24%35%3%5%
  10%32%57%29%45%26%28%74%24%61%
%46%15%7%7119%74%38%9%19%34%256%
%9%11%84%8971%23%28%12%20%35%843%
```

Capítulo 17

Lista de Datasets (LIST)

17.1 Descrição Geral

LIST pode ser usado para imprimir valores de dados de um arquivo, variáveis recodificadas e informação do dicionário IDAMS associado. Variáveis específicas podem ser selecionadas para impressão, ou os dados inteiros e/ou dicionário pode ser listado.

Cada registro em um arquivo de dados é um fluxo contínuo de valores de dados. Quando impressos como estão, se torna difícil distingüir os valores de variáveis adjacentes. LIST elimina este incoveniente ao oferecer formato de impressão de dados que separa os valores das variáveis.

Um dicionário do IDAMS pode ser impresso sem o arquivo Dados correspondente ao fornecer-se um arquivo dummy (i.e. um arquivo nulo ou vazio), ao definir-se o arquivo Dados.

17.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Casos podem ser selecionados utilizando-se um filtro, ou a opção de pular casos (SKIP). A opção de pular, se usada, especifica que o primeiro e cada n-ésimo caso subseqüente deve ser impresso. Se um filtro é especificado, a opção de pular se aplica aos casos que passarem no filtro. Dos casos selecionados, os valores de dados são listados para todas as variáveis descritas no dicionário ou um subconjunto se o parâmetro VARS é especificado.

Transformando dados. Declarações de Recode podem ser usadas.

Tratamento de dados perdidos. Valores de dados perdidos são impressos quando eles ocorrem, não causando ação especial.

17.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, apenas para variáveis usadas na execução. Se todas as variáveis são selecionadas para impressão, então o dicionário completo é impresso em ordem seqüencial.

Dados. Variáveis numéricas são impressas com ponto decimal explícito, se houver, e sem zeros iniciais. Se um valor ultrapassa a largura de campo, ele é impresso como uma string de asteriscos. Dados ruins substituídos por códigos de dados perdidos de default são impressos como espaços em branco. Valores para uma variável são impressos em uma coluna que se extende pelas páginas, para todos os casos selecionados para impressão. Abaixo segue um esboço em um bloco do formato de impressão:

```
  v      v   v          v
xxx   xxxx   x   xxxxxxxx
xxx   xxxx   x   xxxxxxxx
xxx   xxxx   x   xxxxxxxx
 .      .    .       .
 .      .    .       .
```

Os cabeçalhos de v nas colunas representam os números de variáveis e os x's representam os valores de variável. Se o usuário requisita impressão de mais variáveis do que pode caber na linha (127 caracteres), LIST fará um número de passagens pelos dados, listando tantas variáveis quanto possíveis de cada vez. Por exemplo, se 50 variáveis devem ser impressas, LIST lerá os dados, imprimindo todos os valores, digamos, para as primeiras 10 variáveis. Então os dados serão lidos novamente para a impressão, digamos para as próximas 12 variáveis, e assim por diante. O número de variáveis impressas em cada passagem ao longo dos dados depende da largura do campo das variáveis sendo impressas e é automaticamente computado pelo LIST.

Seqüência e identificação de casos. Existem opções para imprimir um número de seqüência de caso e/ou valores de variáveis de identificação com cada caso. (Ver os parâmetros PRINT e IDVARS). Eles são impressos como as primeiras colunas.

Variáveis de Recode. Elas são impressas com 11 dígitos incluindo um ponto decimal explícito e 2 casas decimais.

17.4 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário IDAMS. Se apenas uma listagem do dicionário é requerida, o arquivo Dados é especificado como NUL.

17.5 Estrutura de Setup

```
$RUN LIST

$FILES
     Especificações de arquivo

$RECODE (optional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados

Files:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default  IDAMS.LST)
```

17.6 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, ítens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V5=100-199
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  PRINTING THE STUDY: 113A
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  VARS=(V3,V10-V25) IDVARS=V1
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

   MAXCASES=n
   O número máximo de casos a serem impressos.
   Default: Todos os casos serão impressos.

   SKIP=n
   Cada n-ésimo caso (ou cada n-ésimo caso passando no filtro) é impresso, começando com o primeiro caso. O último caso será sempre impresso, a não ser que a opção MAXCASES proíba isso.
   Default: Todos os casos (ou todos os casos passando pelo filtro) são impressos.

   VARS=(lista de variáveis)
   Imprime os valores de dados para as variáveis especificadas. Valores de variáveis serão impressos na ordem em que eles aparecem na nessa lista.
   Default: Todas as variáveis no diconário são listadas.

   IDVARS=(lista de variáveis)
   Os valores das variáveis especificadas são impressos para identificar cada caso.

   SPACE=**3**/n
   Número de espaços entre colunas.
   O valor máximo é SPACE=8.

   PRINT=(CDICT/DICT, SEQNUM, **LONG**/SHORT, **SINGLE**/DOUBLE)

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
SEQN	Imprime um número de seqüência de caso para cada caso impresso. Note que casos são numerados depois do filtro ser aplicado.
LONG	Assuma 127 caracteres por linha de impressão.
SHOR	Assuma 70 caracteres por linha de impressão.
SING	Espaço simples entre linhas de dados.
DOUB	Espaço duplo entre linhas de dados.

17.7 Restrição

A soma das larguras dos campos das variáveis a serem impressas, incluindo variáveis de ID de casos, deve ser menor ou igual a 10.000 caracteres.

17.8 Exemplos

Exemplo 1. Listando cinqüenta variáveis incluindo uma variável recodificada; todos os casos serão impressos com suas variáveis de identificação (V1, V2 e V4); o dicionário será impresso, mas sem registros-C.

```
$RUN LIST
$FILES
PRINT  = LIST1.LST
DICTIN = STUDY.DIC                      arquivo Dicionário de entrada
DATAIN = STUDY.DAT                      arquivo Dados de entrada
$RECODE
      R6=BRAC(V6,0-50=1,51-99=2)
$SETUP
LISTING THE VALUES OF 50 VARIABLES WITH 3 ID VARIABLES WITH EACH GROUP
IDVA=(V1,V2,V4)  VARS=(V3-V49,V59,V52,R6)  PRIN=DICT
```

Exemplo 2. Listando um dicionário completo com registros-C sem listar os dados.

```
$RUN LIST
$FILES
DICTIN = STUDY.DIC                      arquivo Dicionário de entrada
DATAIN = NUL
$SETUP
LISTING COMPLETE DICTIONARY
PRIN=CDICT
```

Exemplo 3. Checar a recodificação pela listagem de valores de variáveis de entrada e recodificadas para 10 casos.

```
$RUN LIST
$FILES
DICTIN = A.DIC                          arquivo Dicionário de entrada
DATAIN = A.DAT                          arquivo Dados de entrada
$RECODE
      R101=COUNT(1,V40-V49)
      IF MDATA(V9,V10) THEN R102=99 ELSE R102=V9+V10
      R103=BRAC(V16,15-24=1,25-34=2,35-54=3,ELSE=9)
$SETUP
CHECKING VALUES FOR 3 RECODED VARIABLES
MAXCASES=10  SKIP=10  SPACE=1  -
   VARS=(V40-V49,R101,V9,V10,R102,V16,R103)
```

Capítulo 18

Intercalação de Datasets (MERGE)

18.1 Descrição Geral

MERGE funde variáveis de casos em um dataset do IDAMS com variáveis de um segundo dataset, juntando os casos de par em par de acordo com uma(s) variável(eis) de emparelhamento. Os casos nos dois datasets não necessitam ser idênticos; isto é, todos os casos presentes em um dataset não precisam estar presentes no outro. O arquivo de dados de saída consiste de registros contendo variáveis especificadas pelo usuário de cada um dos dois arquivos de entrada juntamente com um dicionário IDAMS correspondente. Para distinguir os dois datasets, um é referido como "dataset A", e o outro "dataset B" ao longo do write-up.

Combinando datasets com coleções idênticas de casos. Um exemplo de um uso do programa é a combinação de dados de uma primeira e uma subsequente onda de entrevistas com a mesma coleção de respondentes.

Combinando datasets com coleções de casos de dados ligeiramente diferentes. Quando há mais de uma onda de entrevistas em um survey, alguns respondentes podem desaparecer, e outros podem ser adicionados. O programa permite essas discrepâncias entre datasets e pode, por exemplo, ser requerido a produzir os registros para todos os respondentes, incluindo aqueles entrevistados em apenas uma onda. Nesse exemplo, os valores das variáveis para a onda onde o respondente não foi entrevistado seriam processados como valores de dados perdidos.

Combinando datasets com diferentes níveis de dados. MERGE pode também ser usado para combinar dois datasets, um dos quais contém dados a um nível mais agregado do que o outro. Por exemplo, dados de domicílio podem ser adicionados a registros individuais dos membros do domicílio.

18.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Um filtro pode ser especificado para apenas um dos datasets ou para ambos. A única diferença no formato do filtro é que ele deve ser precedido por um "A:" ou "B:" nas colunas 1-2 para indicar o dataset onde o filtro se aplica.

Todas as variáveis selecionadas de cada dataset podem ser incluídas no dataset de saída. Essas variáveis de saída são especificadas em uma lista de variáveis que possui o formato usual, exceto que as variáveis são denotadas por um "A" ou "B" (ao invés de "V") para identificar o dataset de entrada onde elas existem. Por exemplo, "A1, B5, A3-A45" seleciona variáveis V1, V3-V45 do dataset A e variável V5 do dataset B. Ver a descrição de variáveis de saída na seção "Declarações de Controle de Programa".

Transformando dados. Declarações de Recode não podem ser utilizadas.

Tratamento de dados perdidos. Para as opções MATCH=UNION, MATCH=A, e MATCH=B, códigos de dados perdidos são usados como valores para as variáveis de saída que não estejam disponíveis para um caso particular. Ver o parágrafo "Manuseando casos que aparecem em apenas um dataset de entrada" na seção descrevendo o dataset de saída abaixo. Os códigos de dados perdidos são obtidos dos dicionários dos datasets A e B. O usuário especifica, para cada dataset, se o primeiro ou segundo código de dados perdidos

deve ser usado, e isso para todas as variáveis do dataset (ver os parâmetros APAD e BPAD). Se uma variável não tem um código de dados perdidos apropriado no dicionário, então espaços em branco serão utilizados.

Dados perdidos nunca são produzidos como o valor de uma variável de saída que é também um das variáveis de emparelhamento, porque o valor de uma variável de emparelhamento está sempre disponível naquele dataset que contém o caso. Por exemplo, com MATCH=UNION selecionado, suponha que a variável A1 e B3 foram usadas como variáveis de emparelhamento e que somente A1 foi listada como uma variável de saída (A1 e B3 não seriam listadas simultâneamente, pois presumivelmente elas possuem o mesmo valor): então, se um caso no dataset A era perdido, o valor da variável de saída A1 seria o valor de B3.

18.3 Resultados

Números de variáveis anteriores (entrada) versus novos (saída). (Opcional: ver o parâmetro PRINT). Um quadro contendo os números de variáveis de entrada e números de referências, e os números de variáveis correspondentes e números de referências.

Dicionário de saída. (Opcional: ver o parâmetro PRINT).

Documentação de casos não emparelhados entre os datasets A ou B. Há várias maneiras que casos não emparelhados, i.e. casos aparecendo em apenas um arquivo, podem ser documentados (ver o parâmetro PRINT).

- Os valores de variáveis de emparelhamento podem ser impressos:
  - toda vez que variáveis de saída de um dos datasets forem preenchidas com dados perdidos,
  - toda vez que casos do dataset A forem deletados,
  - toda vez que casos do dataset B forem deletados.
- Os valores de variáveis A podem ser impressos toda a vez que um caso do dataset A não emparelhar com qualquer caso de B. As variáveis são impressas na ordem especificada para o dataset nas variáveis de saída, seguidas por todas as variáveis de emparelhamento que também não são variáveis de saída.
- Os valores de variáveis B podem ser impressos toda a vez que um caso do dataset B não emparelhar com qualquer caso de A. As variáveis são impressas na ordem especificada para o dataset nas variáveis de saída, seguidas por todas as variáveis de emparelhamento que também não são variáveis de saída.

Contagem de casos. O program imprime o número de casos existentes nos datasets A e B, número de casos existentes no dataset A e não em B, número de casos existentes no dataset B e não em A, e o número total de casos de saída escritos.

18.4 Dataset de Saída

A saída é um novo arquivo Dados e um dicionário do IDAMS correspondente.

Cada registro de dados contém os valores das variáveis de saída para casos emparelhados dos datasets A e B. Note que uma variável de emparelhamento não é automaticamente produzida: o usuário deve incluir as variáveis de emparelhamento de um dos datasets na lista de variáveis de saída para dar ao saída uma ID de caso.

Manuseando casos que aparecem em apenas um dataset de entrada. Quatro ações são possíveis:

1. MATCH=INTERSECTION. Casos que aparecem em apenas um dataset não são incluídos no dataset de saída. (Se os datasets A e B são interpretados como conjunto de casos, o saída é a interseção entre A e B).
2. MATCH=UNION. Qualquer caso que aparece em qualquer dataset é incluído no dataset de saída. Variáveis do dataset de entrada que não contém o caso recebem valores de dados perdidos no dataset de saída. (O saída é a união dos conjuntos A e B).

3. MATCH=A. Qualquer caso que aparece no dataset A é incluído no dataset de saída, enquanto um caso que aparece apenas no dataset B não é incluído. Se um caso é encontrado apenas no dataset A, variáveis do dataset B recebem valores de dados perdidos no dataset de saída para aquele caso. (O saída é o conjunto A).

4. MATCH=B. O mesmo que a opção 3, exceto que o dataset B define os casos incluídos no dataset de saída. (O saída é o conjunto B).

Manuseando casos duplicados. Quando um dos dois datasets de entrada contém mais de um caso com o mesmo valor nas variáveis de emparelhamento, o dataset é dito conter casos duplicados. Normalmente (i.e. quando o parâmetro DUPBFILE não é especificado) o programa imprime uma mensagem sobre a ocorrência de duplicatas e, então, trata cada uma delas como um caso separado. Os casos realmente escritos no arquivo de saída dependem da opção MATCH selecionada. A próxima figura mostra como isso funciona.

Fundindo arquivos com duplicatas (DUPBFILE não especificado)

```
    Enrtada                                    Saída

  A     |   B     | MATCH = UNION|  MATCH = A   |   MATCH = B   | MATCH = INTER
        |         |              |              |               |
ID  N1  | ID  N2  | ID   N1   N2 | ID  N1   N2  | ID  N1   N2   | ID  N1   N2
        |         |              |              |               |
01 MARY| 01 JOHN | 01 MARY JOHN | 01 MARY JOHN | 01 MARY JOHN | 01 MARY JOHN
01 ANN | 02 PETER| 01 ANN  ____ | 01 ANN  ____ | 02 JANE PETER| 02 JANE PETER
02 JANE| 03 MIKE | 02 JANE PETER| 02 JANE PETER| 03 ____ MIKE |
       |         | 03 ____ MIKE |              |               |
```

No entanto, duplicatas podem ser interpretadas e manuseadas diferentemente quando um dos dois datasets contém casos a um nível menor de análise do que o outro. Por exemplo, um dataset contém dados de domicílio e o segundo contém dados dos membros do domicílio. Nessa instância, as variáveis de emparelhamento especificadas de cada arquivo seriam a identificação do domicílio. Portanto, "duplicatas" ocorreriam naturalmente no dataset "membro do domicílio", pois muitos domicílios terão mais de um membro. Ao especificar o parâmetro DUPBFILE, a mensagem a respeito da ocorrência de duplicatas não é impressa e casos são construídos para cada caso de "duplicata" no dataset B com as variáveis do caso que serve de emparelhamento em A copiado em cada um. A figura seguinte mostra um exemplo desse procedimento.

Fundindo arquivos em diferentes níveis (DUPBFILE especificado)

```
    Entrada                                    Saída

  A     |    B    | MATCH = UNION|  MATCH = A   |  MATCH = B   | MATCH = INTER
        |         |              |              |              |
ID  N1  | ID  N2  | ID   N1  N2  | ID N1   N2   | ID N1   N2   | ID N1   N2
        |         |              |              |              |
01 JONE| 01 MARY | 01 JONE MARY | 01 JONE MARY | 01 JONE MARY | 01 JONE MARY
03 SMIT| 01 JOHN | 01 JONE JOHN | 01 JONE JOHN | 01 JONE JOHN | 01 JONE JOHN
04 SCOT| 01 ANN  | 01 JONE ANN  | 01 JONE ANN  | 01 JONE ANN  | 01 JONE ANN
       | 02 PETE | 02 ____ PETE | 03 SMIT MIKE | 02 ____ PETE | 03 SMIT MIKE
       | 02 JANE | 02 ____ JANE | 04 SCOT ____ | 02 ____ JANE |
       | 03 MIKE | 03 SMIT MIKE |              | 03 SMIT MIKE |
       |         | 04 SCOT ____ |              |              |
```

Seqüência de variáveis e número de variáveis. Variáveis são produzidas na ordem que é dada na lista de variáveis de saída e são sempre renumeradas, começando do valor do parâmetro VSTART. Portanto, uma lista de variáveis de saída como "A1-A5, B6, A7-A25, B100" criaria um dataset com variáveis V1 até V26 se VSTART=1. Números de referência para variáveis, se elas existem, são transferidos intactos para o dicionário de saída.

Localizações de variável. Localizações de variável são atribuídas por MERGE, começando com a primeira variável de saída e continuando na ordem da lista de variáveis de saída.

18.5 Dataset de Entrada

MERGE requer 2 arquivos Dados cada um descrito por um dicionário do IDAMS.

As variáveis de emparelhamento podem ser alfabéticas ou numéricas. Variáveis de emparelhamento correspondentes dos datasets A e B devem ter a mesma largura de campo.

As variáveis de saída podem ser alfabéticas ou numéricas.

Cada arquivo Dados de entrada deve ser classificado em ordem crescente das suas variáveis de emparelhamento antes de usar MERGE.

18.6 Estrutura de Setup

```
$RUN MERGE

$FILES
     Especificações de arquivo

$SETUP
     1. Filtro(s) (opcional)
     2. Título
     3. Parâmetros
     4. Especificação de variáveis de emparelhamento
     5. Variáveis de saída

$DICT (condicional)
     Dicionário (ver Nota abaixo)

$DATA (condicional)
     Dados (ver Nota abaixo)


Arquivos:
DICTxxxx   dicionário de entrada para o dataset A
           (omitir se $DICT não é usado)
DATAxxxx   dados de entrada para o dataset A
           (omitir se $DATA não é usado)
DICTyyyy   dicionário de entrada para o dataset B
           (omitir se $DICT não é usado)
DATAyyyy   dados de entrada para o dataset B
           (omitir se $DATA não é usado)
DICTzzzz   dicionário de saída
DATAzzzz   dados de saída
PRINT      resultados (default  IDAMS.LST)
```

Nota. O dataset A ou o B, mas não ambos, pode ser introduzido no setup. Contudo, registros seguindo $DICT e $DATA são copiados em arquivos definidos por DICTIN e DATAIN, respectivamente. Portanto, se o arquivo A é introduzido no setup, o dataset A será definido por DICTIN e DATAIN e INAFILE=IN deve ser especificado. Similarmente, se o arquivo B é introduzido no setup, então INBFILE=IN deve ser especificado.

18.7 Declarações de Controle de Programa

Refira-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, ítens 1-3 abaixo.

1. **Filtro(s)** (opcional). Seleciona um subconjunto de casos de um dataset A e/ou B a ser usado na execução. Note que cada declaração de filtro deve ser precedida por "A:" ou "B:" nas colunas um e dois para indicar o dataset onde o filtro deve ser aplicado.

```
Exemplo: A: INCLUDE V1=10,20,30
         B: INCLUDE V1=10,20,30
```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

```
Exemplo: MERGE OF TEACHER DATA AND STUDENT DATA
```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

```
Exemplo: MATCH=INTE PRINT=(A, B)
```

INAFILE=**INA**/xxxx
: Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada de A.
Default ddnames: DICTINA, DATAINA.

INBFILE=**INB**/xxxx
: Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada de B.
Default ddnames: DICTINB, DATAINB.

MAXCASES=n
: O número máximo de casos (depois da filtragem) a ser usado do arquivo de entrada de A.
Default: Todos os casos serão utilizados.

MATCH=INTERSECTION/UNION/A/B

INTE	Processa apenas casos aparecendo em ambos os datasets A e B.
UNIO	Processa casos aparecendo em apenas um ou em ambos os datasets A e B, preenchendo com dados perdidos quando necessário.
A	Processa casos aparecendo apenas no dataset A, preenchendo as de B com dados perdidos quando necessário.
B	Processa casos aparecendo apenas no dataset B, preenchendo as de A com dados perdidos quando necessário.

Não há default.

DUPBFILE
: Um caso no dataset A pode ser emparelhado com um ou mais casos (i.e. duplicatas) do dataset B. Para cada emparelhamento, um registro de saída será criado, dependendo do parâmetro MATCH.
Nota: O dataset com as duplicadas esperadas deve ser definido como o dataset B.
Default: Casos duplicados em qualquer dataset serão detectados no resultado impresso e, então tratados como casos distintos de acordo com a especificação do MATCH.

OUTFILE=**OUT**/zzzz
: Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
Default ddnames: DICTOUT, DATAOUT.

VSTART=**1**/n
: Número da variável para a primeira variável no dataset de saída.

APAD=**MD1**/MD2

Ao preencher variáveis de A com dados perdidos:

MD1 Produza o primeiro código de dados perdidos.

MD2 Produza o segundo código de dados perdidos.

BPAD=**MD1**/MD2

Ao preencher variáveis de B com dados perdidos:

MD1 Produza o primeiro código de dados perdidos.

MD2 Produza o segundo código de dados perdidos.

PRINT=(**PAD**/NOPAD, **ADELETE**/NOADELETE, **BDELETE**/NOBDELETE, VARNOS, A, B, **OUTDICT**/OUTCDICT/NOOUTDICT)

PAD Imprima os valores das variáveis de emparelhamento ao preencher quaisquer variáveis A ou B com dados perdidos.

ADEL Imprima os valores das variáveis de emparelhamento para o dataset A toda vez que um caso de A não estiver incluído no arquivo de dados de saída.

BDEL Imprima os valores das variáveis de emparelhamento para o dataset B toda vez que um caso de B não estiver incluído no arquivo de dados de saída.

VARN Imprima uma lista de números de variáveis nos datasets de entrada e os correspondentes números de variáveis nos datasets de saída.

A Imprima todos os resultados e valores de variáveis de emparelhamento para casos aparecendo apenas no dataset A, estando ou não incluídos no dataset de saída.

B Imprima todos os resultados e valores de variáveis de emparelhamento para casos aparecendo apenas no dataset B, estando ou não incluídos no dataset de saída.

OUTD Impima o dicionário de saída sem registros-C.

OUTC Impima o dicionário de saída com registros-C, se houver.

NOOU Não imprima o dicionário de saída.

4. **Especificação de variáveis de emparelhamento** (mandatório). Essa declaração define as variáveis dos datasets A e B que devem ser comparadas para emparelhar casos. Note que cada arquivo de dados de entrada deve ser classificado de acordo com suas variáveis de emparelhamento antes de se usar MERGE.

```
Exemplo:  A1=B3, A5=B1
```

o que significa que para que um caso do dataset A case com um caso do dataset B, o valor da variável V1 de A deve ser idêntico ao valor da variável V3 de B, similarmente para as variáveis V5 e V1.

Formato geral

An=Bm, Aq=Br, ...

Regras de codificação

- A largura do campo das variáveis a serem comparadas deve ser o mesmo. A comparação é feita tomando como base os caracteres, não os números. Portanto, '0.9' não é equivalente a '009', nem é '9' igual a '09'. Se as larguras dos campos não são as mesmas, use o programa TRANS para mudar a largura de uma das variáveis antes de usar MERGE.
- Cada par de variáveis de emparelhamento é separado por uma vírgula.
- Espaços em branco podem ocorrer em qualquer lugar na declaração.
- Para continuar em uma outra linha, termine a informação em um ponta e entre com um traço (-) para indicar continuação.

5. **Variáveis de saída** (mandatório). Isso define quais variáveis de cada dataset de entrada devem ser transferidas para o saída e especifica o ordenamento no saída.

   ```
   Exemplo:  A1, B2, A5-A10, B5, B7-B10
   ```

   o que significa que o dataset de saída conterá a variável V1 do dataset A, seguida pela variável V2 de B, seguida pelas variáveis V5 até V10 do dataset A, etc., nessa ordem.

   Regras de codificação

   - As regras de codificação são as mesmas das de especificação de variáveis com os parâmetros VARS, exceto que A's e B's são usados ao invés de V's. Cada número de variável do dataset A é precedido por um "A" e cada número de variável do dataset B é precedido por um "B".
   - Variáveis duplicadas na lista contam como variáveis separadas.

18.8 Restrições

1. O número máximo de variáveis de emparelhamento de cada dataset é 20.
2. Variáveis de emparelhamento devem ser do mesmo tipo e ter a mesma largura de campo em cada arquivo.
3. O comprimento máximo total do conjunto de variáveis de emparelhamento para cada dataset é de 200 caracteres.

18.9 Exemplos

Exemplo 1. Combinando registros de 2 datasets com um conjunto de casos idênticos; em ambos os datasets, casos são identificados pelas variáveis 1 e 3; todas as variáveis devem ser selecionadas de cada dataset de entrada.

```
$RUN MERGE
$FILES
DICTOUT = AB.DIC          arquivo Dicionário de saída
DATAOUT = AB.DAT          arquivo Dados de saída
DICTINA = A.DIC           arquivo Dicionário de entrada para o dataset A
DATAINA = A.DAT           arquivo Dados de entrada para o dataset A
DICTINB = B.DIC           arquivo Dicionário de entrada para o dataset B
DATAINB = B.DAT           arquivo Dados de entrada para o dataset B
$SETUP
COMBINING RECORDS FROM 2 DATASETS WITH AN IDENTICAL SET OF CASES
MATCH=UNION
A1=B1,A3=B3
A1-A112,B201-B401
```

Exemplo 2. Combinando datasets com coleções de casos ligeiramente diferentes; apenas casos que possuem registros em ambos os datasets são processados; casos são identificados pelas variáveis 2 e 4 no primeiro dataset, e pelas variáveis 105 e 107 respectivamente no segundo dataset; variáveis no dataset de saída serão renumeradas começando do número 201, e uma listagem de referências é requisitada; apenas variáveis selecionadas serão retiradas de cada dataset de entrada.

```
$RUN MERGE
$FILES
      como no Exemplo 1
$SETUP
COMBINING RECORDS FROM 2 DATASETS WITH DIFFERENT SETS OF CASES
MATCH=INTE  VSTA=201  PRIN=VARNOS
A2=B105,A4=B107
B105,B107,A36-A42,B120,B131
```

Exemplo 3. Combinando dataset com níveis diferentes de dados; casos do dataset A são combinados com um subconjunto de casos do dataset B; um caso de A pode ser emparelhado com um ou mais casos de B; casos em A que não fazem o emparelhamento com um caso em B serão descartados e não serão listados.

```
$RUN MERGE
$FILES
      como no Exemplo 1
$SETUP
B:  INCLUDE V18=2 AND V21=3
COMBINING 2 DATASETS WITH DIFFERENT LEVELS OF DATA
MATCH=B  DUPB
A1=B15
B15,A2,A6-A12,B20-B31,B40
```

Exemplo 4. Renda domiciliar deve ser calculada de um arquivo de membros do domicílio e então fundido de volta em registros de membros individuais; AGGREG é usado primeiro para somar renda (V6) dos indivíduos do domicílio; V3 é a variável que identifica o domicílio; o arquivo de saída de AGGREG (definido por DICTAGG e DATAAGG) conterá 2 variáveis, a ID do domicílio (V1) e renda do domicílio (V2); esse arquivo é então usado como o arquivo "A" com MERGE para adicionar a renda domiciliar adequada (variável A2) a cada registro individual original (variáveis B1-B46).

```
$RUN AGGREG
$FILES
PRINT   = MERGE4.LST
DICTIN  = INDIV.DIC           arquivo Dicionário de entrada
DATAIN  = INDIV.DAT           arquivo Dados de entrada
DICTAGG = AGGDIC.TMP          arquivo Dicionário de saída temporário do AGGREG
DATAAGG = AGGDAT.TMP          arquivo Dados de saída temporário do AGGREG
DICTOUT = INDIV2.DIC          arquivo Dicionário de saída do MERGE
DATAOUT = INDIV2.DAT          arquivo Dados de saída do MERGE
$SETUP
AGGREGATING INCOME
IDVARS=V3  AGGV=V6  STATS=SUM  OUTF=AGG
$RUN MERGE
$SETUP
MERGING HOUSEHOLD INCOME TO INDIVIDUAL RECORDS
INAFILE=AGG  INBFILE=IN  DUPB  MATCH=B
A1=B3
B1-B46,A2
```

Note que ao se fazer indicação de arquivos com $FILES, eles não precisam ser repetidos se eles forem reutilizados em passos subseqüentes.

Capítulo 19

Classificação e Fusão de Arquivos (SORMER)

19.1 Descrição Geral

SORMER permite que o usuário execute de maneira mais conveniente uma Classificação/Fusão ao permitir a especificação da informação de campo-de-controle da classificação ou fusão no formato de parâmetro do IDAMS usual. Se o arquivo de dados é descrito por um dicionário do IDAMS, então uma cópia do dicionário correspondendo aos dados classificados pode ser produzida e os campos de classificação podem ser especificados ao fornecer as variáveis apropriadas; caso contrário, elas são especificadas pela sua localização.

Ordem de classificação. O usuário pode especificar que os dados devem ser classificados/fundidos em ordem ascendente ou descendente.

19.2 Características Padrão do IDAMS

SORMER é um programa de utilidade e não contém qualquer das características padrão do IDAMS.

19.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, para variáveis-chave de classificação.

Resultados de classificação/fusão. Número de registros classificados/fundidos.

19.4 Dicionário de Saída

Uma cópia do dicionário de entrada correspondendo ao arquivo Dados de saída.

19.5 Dados de Saída

A saída consiste de um arquivo com os mesmos atributos dos arquivos de entrada com os registros classificados dentro da ordem requisitada.

19.6 Dicionário de Entrada

Se os campos de classificação estão sendo especificados com números de variáveis, então um dicionário do IDAMS contendo registros-T, pelos menos para essas variáveis, deve ser utilizado como entrada. Apenas dicionários descrevendo dados com um registro por caso são permitidos.

19.7 Dados de Entrada

Para **classificação**, um arquivo de dados é usado como entrada, contendo um ou mais campos (ou variáveis) cujos valores definem a ordem desejada.

Para **fusão**, a entrada consiste de 2-16 arquivos de dados, cada um com o mesmo formato de registro, i.e. o mesmo comprimento de registro e campos definindo a ordem de classificação nas mesmas posições. Cada arquivo deve ser classificado na ordem pelos campos de controle de fusão antes da fusão.

19.8 Estrutura de Setup

```
$RUN SORMER

$FILES
     Especificações de arquivo

$SETUP
     1. Título
     2. Parâmetros

$DICT (condicional)
     Dicionário para variáveis de campo de classificação/fusão


Arquivos para classificação:
DICTxxxx  dicionário do IDAMS para variáveis de campo de
          classificação (omitir se $DICT é usado)
SORTIN    dados de entrada
DICTyyyy  dicionário de saída
SORTOUT   dados de saída

Arquivos para a fusão:
DICTxxxx  dicionário do IDAMS para variáveis de campo de
          fusão (omitir se $DICT é usado)
SORTIN01  1° arquivo de dados
SORTIN02  2° arquivo de dados
   .
   .
DICTyyyy  dicionário de saída
SORTOUT   dados de saída

PRINT     resultados (default  IDAMS.LST)
```

Nota. Quando a execução do SOMER é requisitada mais de uma vez em um arquivo de setup, as definições do arquivo de entrada especificadas na execução subseqüente apenas modifica, mas não substiuti, as definições de arquivo de entrada especificadas previamente, e.g. se SORTIN01, SORTIN02 e SORTIN03 são especificados para a primeira execução, e SORTIN01 e SORTIN02 são especificados para a segunda execução no mesmo setup, o 'novo' SORTIN01 e SORTIN02, como também o 'velho' SORTIN03 serão tomados para a fusão.

19.9 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-2 abaixo.

1. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear resultados.

   ```
   Exemplo: SORTING WAVE ONE
   ```

2. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  KEYVARS=(V2,V3)
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para o arquivo Dicionário de entrada.
   Default ddname: DICTIN.

   OUTFILE=yyyy
   Um sufixo ddname de 1-4 caracteres para o arquivo Dicionário de saída.
   Precisa ser especificado para obter em saída uma cópia do Dicionário de entrada.

   SORT/MERGE
   SORT Os dados de entrada devem ser classificados.
   MERG Dois ou mais arquivos de dados devem ser fundidos.

   ORDER=**A**/D
   A Classificar em ordem ascendente nos campos de classificação.
   D Classificar em ordem descendente.

   KEYVARS=(lista de variáveis)
   Lista das variáveis a serem usadas como campos de classificação (dicionário do IDAMS deve ser fornecido).
   Nota: O arquivo de dados deve conter um registro por caso para que essa opção seja selecionada. Se há mais de um registro por caso, então selecione KEYLOC.

   KEYLOC=(s1,e1, s2,e2, ...)
   Sn Localização inicial no n-ésimo campo de classificação.
   En Localização final do n-ésimo campo de classificação. Deve ser especificado mesmo quando igual a localização inicial.

   Nota. Não há defaults. Ou KEYVARS ou KEYLOC (mas não ambos) deve ser especificado.

   PRINT=CDICT/DICT
   CDIC Imprime o dicionário de entrada para as variáveis-chave de classificação com registros-C, se houver.
   DICT Imprime o dicionário de entrada sem registros-C.

19.10 Restrições

1. Um máximo de 16 arquivos podem ser fundidos.
2. Um máximo de 12 campos de controle de Classificação/Fusão ou variáveis podem ser especificadas.
3. O número máximo de registros depende do espaço de disco disponível para os arquivos de trabalho SORTWK01, 02, 03, 04, 05. Esses arquivos de trabalho podem ser direcionados para outro disco que não o disco default, se necessário.

19.11 Exemplos

Exemplo 1. Fundindo três arquivos de dados pré-classificados do mesmo formato; cada arquivo é descrito pelo mesmo dicionário do IDAMS; casos são classificados em ordem ascendente nas três variáveis: V1, V2 e V4.

```
$RUN SORMER
$FILES
PRINT    = SORT1.LST
DICTIN   = \SURV\DICT.DIC          arquivo Dicionário de entrada
SORTIN01 = DATA1.DAT               arquivo Dados de entrada 1
SORTIN02 = DATA2.DAT               arquivo Dados de entrada 2
SORTIN03 = DATA3.DAT               arquivo Dados de entrada 3
DICTOUT  = \SURV\DATA123.DIC       arquivo Dicionário de saída
SORTOUT  = \SURV\DATA123.DAT       arquivo Dados de saída
$SETUP
MERGING THREE IDAMS DATA FILES:  DATA1, DATA2  AND DATA3
MERG  KEYVARS=(V1,V2,V4) OUTF=OUT
```

Exemplo 2. Classificando um arquivo Dados em ordem descendente em dois campos; o primeiro campo tem um comprimento de 4 caracteres, começando na coluna 12; o segundo campo tem um comprimento de 2 caracteres, começando na coluna 3; um dicionário não é utilizado.

```
$RUN SORMER
$FILES
SORTIN  = RAW.DAT              arquivo Dados de entrada
SORTOUT = SORT.DAT             arquivo Dados de saída
$SETUP
SORTING DATA FILE WITHOUT USING DICTIONARY
KEYLOC=(12,15,3,4)  ORDER=D
```

Capítulo 20

Subdivisão de Datasets (SUBSET)

20.1 Descrição Geral

SUBSET coleta subconjuntos de arquivos Dados e dicionário do IDAMS correspondente por caso e/ou por variável, ou copia os arquivos completos.

Checagem de ordem de classificação. O programa possui uma opção para checar se os casos de dados estão em ordem ascendente, baseado em uma lista de variáveis de ordem de classificação (ver o parâmetro SORTVARS). Casos adjacentes com identificação duplicada não são considerados fora de ordem. Contudo, há uma opção para deletar ocorrências duplicadas de qualquer caso.

20.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Subdivisão casos é alcançado ao usar-se um filtro para selecionar um conjunto de casos particular do dataset de entrada. Seleção de variáveis é feita definindo um conjunto de variáveis de entrada a ser transferido para o dataset de saída. As variáveis podem ser processadas em qualquer ordem, e podem ser transferidas mais de uma vez, desde que os números das variáveis de saída sejam re-numerados.

Transformando dados. Declarações de Recode não podem ser utilizadas.

Tratamento de dados perdidos. SUBSET não faz diferença entre valores de dados perdidos e substantivos; todos os dados são tratados da mesma maneira.

20.3 Resultados

Dicionário de saída. (Opcional: ver o parâmetro PRINT).

Estatísticas do subdivisão. O comprimento de registro de saída, o número de registros de dicionário e de dados de saída.

Números de variáveis anteriores (entrada) versus novos (saída). (Opcional: ver o parâmetro PRINT). Um quadro contendo os números de variável de entrada e números de referência, e os números de variáveis de saída correspondentes e números de referência.

Notificação de casos duplicados. (Condicional: se a ordem de classificação do arquivo está sendo checada, todos os casos duplicados são documentados, estando ou não especificado o parâmetro DUPLICATE=DELETE). Para cada identificação de caso que apareça mais de uma vez nos dados, o número de duplicatas, o número seqüencial do caso, e a identificação do caso são impressos. Além disso, o programa imprime o número de registros de dados de entrada e o número de registros de dados de entrada deletados.

20.4 Dataset de Saída

A saída é um dataset do IDAMS construído do subconjunto de casos e/ou variáveis do arquivo de entrada, especificado pelo usuário. Quando todas as variáveis são copiadas, i.e. quando OUTVARS não é especificado, os registros de saída e entrada possuem a mesma estrutura e o dicionário de saída é uma cópia exata do de entrada. Caso contrário, a informação do dicionário para as variáveis no arquivo de saída é designada da seguinte maneira:

Seqüência de variáveis e número de variáveis. Se VSTAR é especificado, variáveis são colocadas como elas aparecem na lista OUTVARS e elas são numeradas de acordo com o parâmetro VSTART. Se VSTART não é especificado, as variáveis de saída assumem os mesmos números das variáveis de entrada e são classificadas em ordem ascendente por número de variável.

Localizações de variável. Localizações de variável são designadas contiguamente de acordo com a ordem das variáveis na lista OUTVARS (se VSTART for especificado) ou depois da classificação em ordem do número da variável (se VSTART não é especificado).

Tipo, largura e número de decimais da variável são os mesmos das variáveis de entrada.

Números de referência. Como no entrada ou modificado de acordo com o parâmetro REFNO.

Registros-C. Códigos e suas labels são copiados como eles estão no dicionário de entrada.

20.5 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Variáveis numéricas ou alfabéticas podem ser usadas.

20.6 Estrutura de Setup

```
$RUN SUBSET

$FILES
     Especificações de arquivo

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída
DATAyyyy   dados de saída
PRINT      resultados (default  IDAMS.LST)
```

20.7 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais profundas das declarações de controle de programa, ítens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V1=10,20,30 AND V2=1,5,7
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  SUBSET OF 1968 ELECTION, V1-V50
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  SORT=(V1,V2), DUPLICATE=DELETE
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   MAXCASES=n
   O número máximo de casos (depois da filtragem) a ser utilizado do arquivo de entrada.
   Default: Todos os casos serão usados.

   SORTVARS=(lista de variáveis)
   Se a ordem de classificação do arquivo deve ser checada, especifique até 20 variáveis que definem a seqüência em ordem maior ou menor. Duplicatas são consideradas como estando em ordem crescente.

   DUPLICATE=**KEEP**/DELETE
   Deleção de casos duplicados (só aplicável se SORT é especificado).
   KEEP Processa todas as ocorrências de casos duplicados.
   DELE Processa apenas a primeira ocorrência de casos duplicados, e imprime mensagem para duplicatas.

   OUTVARS=(lista de variáveis)
   Fornece essa lista somente se um subconjunto de variáveis no dataset de entrada é para ser produzido. Se VSTART não é selecionado, então duplicatas não são permitidas. Caso contrário, variáveis podem ser fornecidas em qualquer ordem e repetidas quando necessário.
   Default: Todas as variáveis são produzidas.

   OUTFILE=**OUT**/yyyy
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
   Default ddnames: DICTOUT, DATAOUT.

   VSTART=n
   As variáveis serão numeradas seqüencialmente, começando em n, no dataset de saída.
   Default: Números de variáveis de entrada são retidos.

   REFNO=**OLDREF**/VARNO
   OLDR Retenha os números de referências em registros-C e -T como no dicionário de entrada.
   VARN Renove o número de referência nos registros-C e -T para fazer o match com o número da variável de saída.

   PRINT=(**OUTDICT**/OUTCDICT/NOOUTDICT, VARNOS)
   OUTD Imprime o dicionário de saída sem registros-C.
   OUTC Imprime o dicionário de saída com registros-C, se houver.
   VARN Imprime uma lista de números anteriores e novos de variáveis e números de referência.

20.8 Restrições

1. O número máximo de variáveis de classificação que podem ser definidas é 20.
2. As larguras combinadas dos campos das variáveis de classificação não deve exceder 200 caracteres.

20.9 Exemplos

Exemplo 1. Construindo um subconjunto de casos para variáveis selecionadas; variáveis serão renumeradas começando em 1 e uma tabela dando os números anteriores e novos de variável serão impressos.

```
$RUN SUBSET
$FILES
PRINT   = SUBS1.LST
DICTIN  = ABC.DIC                       arquivo Dicionário de entrada
DATAIN  = ABC.DAT                       arquivo Dados de entrada
DICTOUT = SUBS.DIC                      arquivo Dicionário de saída
DATAOUT = SUBS.DAT                      arquivo Dados de saída
$SETUP
INCLUDE V5=2,4,5 AND V6=2301
SUBSETTING VARIABLES AND CASES
PRINT=VARNOS  VSTART=1  -
   OUTVARS=(V1-V5,V18,V43-V57,V114,V116)
```

Exemplo 2. Usando o programa SUBSET para checar casos duplicados; casos são identificados por variáveis nas colunas 1-3 e 7-8; há um registro por caso; o dataset de saída não é requerido é não é mantido.

```
$RUN SUBSET
$FILES
DATAIN  = DEMOG.DAT                     arquivo Dados de entrada
$SETUP
CHECKING FOR DUPLICATE CASES
SORT=(V2,V4)  PRIN=NOOUTDICT
$DICT
$PRINT
   3   2   4   1   1
T   2 CASE FIRST ID VAR          1   3
T   4 CASE SECOND ID VAR         7   2
```

Capítulo 21

Transformação de Dados (TRANS)

21.1 Descrição Geral

O programa TRANS cria um novo dataset do IDAMS contendo variáveis de um dataset já existente e novas variáveis definidas por declarações de Recode. É a maneira de salvar variáveis recodificadas.

TRANS possui uma opção de impressão e, portanto, pode ser também utilizado para testar declarações de Recode em um número pequeno de casos antes de executar um programa de análise ou antes de salvar completamente o arquivo.

21.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. Seleção de variáveis é alcançada através do parâmetro OUTVARS.

Transformando dados. Declarações de Recode podem ser usadas.

Tratamento de dados perdidos. Códigos de dados perdidos apropriados são escritos no dicionário de saída; eles são normalmente copiados do dicionário de entrada, mas podem também ser deixados intactos ou fornecidos para variáveis de saída através da declaração de Recode MDCODES. Nenhuma checagem de dados perdidos é feita nos valores de dados exceto através do uso de declarações de Recode.

21.3 Resultados

Dicionário de saída. (Opcional: ver o parâmetro PRINT).

Dados de saída. (Opcional: ver o parâmetro PRINT). Valores para todos os casos de cada variável-V ou -R são dados, 10 valores de variável por linha. Para variáveis alfabéticas, apenas os 10 primeiros caracteres são impressos.

21.4 Dataset de Saída

A saída é um dataset do IDAMS que contém apenas aquelas variáveis (V e R) especificadas no parâmetro OUTVARS. A informação do dicionário para as variáveis no arquivo de saída é designada da seguinte maneira:

Seqüência de variáveis e número de variáveis. Se VSTART é especificado, variáveis são colocadas como elas aparecem na lista OUTVARS e elas são numeradas de acordo com o parâmetro VSTART. Se VSTART não é especificado, as variáveis de saída têm o mesmo número da lista OUTVARS e elas são classificadas em ordem crescente por número de variável.

Nomes de variável e códigos de dados perdidos. Obtidos do dicionário de entrada (apenas variáveis-V) ou das declarações de Recode NAME e MDCODES, se houver.

Localizações de variável. Localizações de variável são designadas contiguamente de acordo com a ordem das variáveis na lista OUTVARS (se VSTART é especificada) ou depois da classificação em ordem de número de variável (se VSTART não é especificada).

Tipo, largura e número de decimais de variável.

Variáveis-V: Tipo, largura de campo e número de casas decimais são os mesmos dos seus valores de entrada.

Variáveis-R: O tipo para variáveis-R é sempre numérico; largura e número de casas decimais são sempre designados de acordo com os valores especificados pelo parâmetro WIDTH (default 9) e DEC (default 0), ou de acordo com os valores fornecidos pelas variáveis individuais nas especificações do dicionário.

Números de referência e ID de estudo. O número de referência e ID de estudo para uma variável-V são sempre os mesmos que seus valores de entrada. Para variáveis-R, o número de referência é deixado em branco e a ID de estudo é sempre REC.

Registros-C. Registros-C não podem ser criados por variáveis-R. Registros-C (se houver) para todas as variáveis-V são copiados no dicionário de saída. Note que se uma variável-V é recodificada durante a execução de TRANS, os registros-C que são produzidos não se aplicam mais à nova versão da variável.

21.5 Dataset de Entrada

A entrada é um arquivo de dados descrito por um dicionário do IDAMS. Variáveis numéricas ou alfabéticas podem ser usadas.

21.6 Estrutura de Setup

```
$RUN TRANS

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de dicionário (opcional)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados

Files:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída
DATAyyyy   dados de saída
PRINT      resultados (default  IDAMS.LST)
```

21.7 Declarações de Controle de Programa

Referir-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-4 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  EXCLUDE V19=2-3
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  CONSTRUCTING VIOLENCE INDICATORS
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  VSTART=1, WIDTH=2  OUTVARS=(V2-V5,R7)
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores de dados de entrada não-numéricos e valores de saída com "largura de campo insuficiente". Ver o capítulo "O Arquivo Setup do IDAMS".

   MAXCASES=n
   O número máximo de casos (depois de filtragem) a ser utilizado do arquivo de entrada.
   Default: Todos os casos serão utilizados.

   MAXERR=**0**/n
   O número máximo de erros de "largura de campo insuficiente" permitido antes da execução parar. Esses erros ocorrem quando o valor de uma variável é grande demais para caber no campo designado, e.g. um valor de 250 quando WIDTH=2 tiver sido especificado. Ver o capítulo "Dados em IDAMS".

   OUTFILE=**OUT**/yyyy
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
   Default ddnames: DICTOUT, DATAOUT.

   OUTVARS=(lista de variáveis)
   Variáveis-V e -R que devem ser produzidas. A ordem das variáveis na lista só será significante se o parâmetro VSTART for especificado. Se VSTART não for especificado todos os números de variável-V e -R devem ser únicos.
   Não há default.

   VSTART=n
   As variáveis serão numeradas seqüencialmente, começando em n, no dataset de saída.
   Default: Números de variável de entrada são retidos.

   WIDTH=**9**/n
   A largura do campo de variável de saída default a ser utilizada para variáveis-R. Esse default pode ser desconsiderado para variáveis específicas com a especificação de dicionário WIDTH. Para mudar a largura de campo de uma variável-V numérica, crie uma variável-R equivalente (ver Exemplo 1).

   DEC=**0**/n
   Número de casas decimais a serem retidas para variáveis-R.

PRINT=(**OUTDICT**/OUTCDICT/NOOUTDICT, DATA)

OUTD Imprima o dicionário de saída sem registros-C.

OUTC Imprima o dicionário de saída com registros-C, se houver.

DATA Imprima os valores das variáveis de saída.

4. **Especificações de dicionário** (opcional). Para qualquer conjunto de variáveis em particular, a largura de campo e o número de casas decimais podem ser especifcados. Essas especificações desconsiderarão os valores colocados pelos parâmetros principais WIDTH e DEC. Note que os códigos de dados perdidos e nomes de variáveis são designados pelas declarações de Recode MDCODES e NAME, respectivamente. Cuidado: A declaração MDCODES retém apenas 2 casas decimais para variáveis-R, arredondando os valores para mais.

   As regras de codificação são as mesmas dos parâmetros. Cada especificação de dicionário deve iniciar em uma nova linha.

```
Exemplos:  VARS=R4, WIDTH=4, DEC=1
           VARS=R8, WIDTH=2
           VARS=(R100-R109), WIDTH=1
```

VARS=(lista de variáveis)

As variáveis-R onde os parâmetros WIDTH e DEC se aplicam.

WIDTH=n

Largura de campo para as variáveis de saída.
Default: Valor dado pelo parâmetro WIDTH.

DEC=n

Número de casas decimais.
Default: Valor dado pelo parâmetro DEC.

21.8 Restrições

1. O número máximo de variáveis-R que pode ser produzido é 250.
2. O número máximo de variáveis que pode ser usado na execução (incluindo variáveis utilizadas somente em declarações de Recode) é 500.
3. O número máximo de especificações de dicionário é 200.

21.9 Exemplos

Exemplo 1. Variáveis selecionadas do dataset de entrada são transferidas para o arquivo de saída juntamente com 2 novas variáveis; números de variável não são mudados; a largura do campo da variável de entrada V20 é mudada para 4.

```
$RUN TRANS
$FILES
PRINT   = TRANS1.LST
DICTIN  = OLD.DIC                        arquivo Dicionário de entrada
DATAIN  = OLD.DAT                        arquivo Dados de entrada
DICTOUT = NEW.DIC                        arquivo Dicionário de saída
DATAOUT = NEW.DAT                        arquivo Dados de saída
$SETUP
CONSTRUCTING TWO NEW VARIABLES
PRINT=NOOUTDICT  OUTVARS=(V1-V19,R20,V33,V45-V50,R105,R122)
VARS=R105,WIDTH=1
VARS=R122,WIDTH=3,DEC=1
VARS=R20,WIDTH=4
$RECODE
   R20=V20
   NAME R20'VARIABLE 20'
   R105=BRAC(V5,15-25=1,<36=2,<46=3,<56=4,<66=5,<90=6,ELSE=9)
   MDCODES R105(9)
   NAME R105'GROUPS OF AGE'
   IF MDATA(V22) THEN R122=99.9 ELSE R122=V22/3
   MDCODES R122(99.9)
   NAME R122'NO ARTICLES PER YEAR'
```

Exemplo 2. Esse exemplo mostra o uso de TRANS para checar declarações de Recode; valores de dados para as variáveis de ID (V1, V2), as variáveis sendo usadas nas recodificações e as variáveis de resultado são listadas para os primeiros 30 casos; o dataset de saída não é requisitado e não é definido.

```
$RUN TRANS
$FILES
PRINT  = TRANS2.LST
DICTIN = STUDY.DIC                       arquivo Dicionário de entrada
DATAIN = STUDY.DAT                       arquivo Dados de entrada
$SETUP
CHECKING RECODES
WIDTH=2  PRINT=(DATA,NOOUTDICT)  MAXCASES=30  -
    OUTVARS=(V1-V2,V71-V74,V118,V12,V13,R901-R903)
$RECODE
   R901=BRAC(V118,1-16=2,17=1,18-23=3,24=1,25-35=3,36=1,37=2,ELSE=9)
   IF NOT MDATA(V12,V13) THEN R902=TRUNC(V12/V13) ELSE R902=99
   R903=COUNT(1,V71-V74)
```

Exemplo 3. Criando um arquivo de teste de dados com uma amostra de 1/20 do arquivo de dados; não há necessidade de se salvar o dicionário de saída, pois será idêntico ao de entrada.

```
$RUN TRANS
$FILES
DICTIN  = STUDY.DIC                      arquivo Dicionário de entrada
DATAIN  = STUDY.DAT                      arquivo Dados de entrada
DATAOUT = TESTDATA                       arquivo Dados de saída
$SETUP
CREATING TEST FILE WITH ALL VARIABLES AND 1/20 SAMPLE OF CASES
PRINT=NOOUTDICT  OUTVARS=(V1-V505)
$RECODE
   IF RAND(0,20) NE 1 THEN REJECT
```

Parte IV

Facilidades para Análise de Dados

Capítulo 22

Análise de Agrupamento (CLUSFIND)

22.1 Descrição Geral

CLUSFIND conduz análise de agrupamento particionando um conjunto de objetos (casos ou variáveis) em um conjunto de clusters determinado por um dos seis algoritmos: dois algoritmos baseados em partição ao redor de medoides, um baseado em agrupamento difuso e três baseados em agrupamento hierárquico.

22.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Se dados brutos são usados como entrada, o filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. As variáveis para análise são especificadas no parâmetro VARS.

Transformando dados. Se dados brutos são usados como entrada, declarações de Recode podem ser utilizadas.

Ponderando dados. Uso de variáveis de ponderação não é aplicável.

Tratamento de dados perdidos. Se dados brutos são usados como entrada, o parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houverem, devem ser usados para checar a existência de dados perdidos. Os casos onde dados perdidos ocorrerem em todas as variáveis serão deletados automaticamente. Caso contrário, dados perdidos são suprimidos "aos pares". Se os dados estão padronizados, a média e o desvio absoluto médio são calculados usando apenas valores válidos. Ao calcular-se as distâncias, apenas aquelas variáveis são consideradas na soma onde valores válidos estejam presentes em ambos os objetos.

Se uma matriz é usada como entrada, o parâmetro MDMATRIX está disponível para indicar que valor deve ser usado para checar a existência de elementos de matriz inválidos.

22.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Variável descritora de registros, e registros-C, se houver, apenas para variáveis usadas na execução.

Dados de entrada depois de padronização. (Opcional: ver o parâmetro PRINT).

Valores padronizados para cada variáveis-R ou -V usadas na análise, precedido pela média e o desvio absoluto médio para aquelas variáveis.

Matriz de dissimilaridade. (Opcional: ver o parâmetro PRINT). A porção triangular inferior esquerda

da matriz, como entrada ou computada pelo programa.

Resultados da análise PAM. Para cada número de clusters da vez (indo de CMIN até CMAX), o seguinte é impresso:

número de objetos representativos (clusters) e a distância média final,
para cada cluster: ID do objeto representativo, número de objetos e a lista de objetos pertencentes a esse cluster,
coordenadas dos medoides (valores das variáveis de análise para cada objeto representativo; para dataset de entrada apenas),
vetor de agrupamento (vetor de números correspondentes aos objetos indicando a que cluster cada objeto pertence) e características de agrupamento,
representação gráfica dos resultados, i.e. uma plotagem da silhueta para cada cluster (opcional - ver o parâmetro PRINT).

Resultados da análise FANNY. Para cada número de clusters da vez (indo de CMIN até CMAX) o seguinte impresso:

número de clusters,
valor da função objetivo a cada iteração,
para cada objeto, a sua ID o coeficiente de filiação para cada cluster,
coeficiente de partição de Dunn e sua versão normalizada,
agrupamento duro mais próximo, i.e. o número de objetos e a lista de objetos pertencendo a cada cluster,
vetor de agrupamento,
representação gráfica dos resultados, i.e. uma plotagem da silhueta para cada cluster (opcional - ver o parâmetro PRINT).

Resultados da análise CLARA. Para o número de clusters experimentados o seguinte é impresso:

lista de objetos selecionados na amostra retida,
vetor de agrupamento,
para cada cluster: ID de objeto representativo, número de objetos e lista de objetos pertencentes a esse cluster,
distância média e máxima a cada medoide,
representação gráfica dos resultados, i.e. uma plotagem da silhueta para cada cluster pertencendo a cada cluster (opcional - ver o parâmetro PRINT).

Resultados da análise AGNES contém o seguinte:

ordenamento final dos objetos (identificados pelas suas ID) e dissimilaridades entre eles,
representação gráfica dos resultados, i.e. uma plotagem de banner de dissimilaridades (opcional - ver o parâmetro PRINT).

Resultados da análise DIANA contém o seguinte:

ordenamento final dos objetos (identificada pelas suas ID) e diâmetros dos clusters,
representação gráfica dos resultados, i.e. uma plotagem de banner de dissimilaridades (opcional - ver o parâmetro PRINT).

Resultados da análise MONA contém o seguinte:

traço de splits (opcional - ver o parâmetro PRINT) com, para cada passo, o cluster a ser separado, a lista de objetos (identificados pelas seus valores de variável de ID) em cada um dos dois subconjuntos e da variável usada para separação,
o ordenamento final dos objetos,
representação gráfica dos resultados, i.e. uma plotagem de separação com a lista de objeto sem cada cluster e a variável usada para separação (opcional - ver o parâmetro PRINT).

22.4 Dataset de Entrada

O dataset de entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis utilizadas na análise devem ser numéricas; elas podem ser inteiras ou com valores decimais. A variável de ID do caso pode ser alfabética. Variáveis usadas na análise PAM, CLARA, FANNY, AGNES ou DIANA devem ter escalas em intervalos. Variáveis usadas na análise MONA devem ser binárias (com valores 0 ou 1). Note que CLUSFIND usa até 8 caracteres do nome da variável como dado no dicionário.

22.5 Matriz de Entrada

Essa é uma matriz quadrada do IDAMS. Ver no capítulo "Dados em IDAMS". Ela pode conter medidas de similaridades, dissimilaridades ou coeficientes de correlação. Note que CLUSFIND usa no máximo 8 caracteres do nome do objeto como dado nos registros de identificação da variável.

22.6 Estrutura de Setup

```
$RUN CLUSFIND

$FILES
     Especificações de arquivo

$RECODE (opcional com dados de entrada brutos; indisponível com entrada de matriz)
     Declarações de Recode

$SETUP
     1. Filtro (opcional; para dados de entrada brutos apenas)
     2. Título
     3. Parâmetros

$DICT (condicional)
     Dicionário para dados de entrada brutos

$DATA (condicional)
     Dados para dados de entrada brutos

$MATRIX (condicional)
     Matriz para entrada de matriz


Arquivos:
FT09       matriz de entrada (se $MATRIX não é usado e entrada de matriz)
DICTxxxx   dicionário de entrada (se $DICT não é usado e INPUT=RAWDATA)
DATAxxxx   dados de entrada (se $DATA não é usado e INPUT=RAWDATA)
PRINT      resultados (default IDAMS.LST)
```

22.7 Declarações de Controle de Programa

Referir-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle do programa, itens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução. Disponível apenas com dados brutos.

   ```
   Exemplo:  INCLUDE V8=5-10
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  PARTITION AROUND MEDOIDS
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  ANALYSIS=PAM   VARS=(V7-V12)  IDVAR=V1
   ```

INPUT=**RAWDATA**/SIMILARITIES/DISSIMILARITIES/CORRELATIONS

RAWD	Entrada: arquivo Dados descrito por um dicionário do IDAMS.
SIMI	Entrada: medidas de similaridades na forma de uma matriz quadrada do IDAMS.
DISS	Entrada: medidas de dissimilaridades na forma de uma matriz quadrada do IDAMS.
CORR	Entrada: coeficientes de correlação na forma de uma matriz quadrada do IDAMS.

Parâmetros apenas para dados de entrada brutos

INFILE=**IN**/xxxx

Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2

Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=**100**/n

O número máximo de casos (depois de filtragem) a ser usado do arquivo de entrada.
Seu valor depende da memória disponível.

n=0	Nenhuma execução, apenas verificação de parâmetros.
0<n<=100	Execução normal.
n>100	Apenas análise CLARA permitida.

MDVALUES=**BOTH**/MD1/MD2/NONE

Quais valores de dados perdidos devem se utilizados para as variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".

STANDARDIZE

Padronizar as variáveis antes de computar as dissimilaridades.

DTYPE=**EUCLIDEAN**/CITY

Tipo de distância a ser usada para computar dissimilaridades.

EUCL	Distância euclidiana.
CITY	Distância city-block.

IDVAR=número de variável

Variável a ser impressa como ID de caso. Apenas três caracteres são utilizados nos resultados. Portanto, variáveis inteiras devem ter valores menores que 1000. Apenas os três primeiros caracteres de uma variável alfabética são impressos.
Não há default.

PRINT=(CDICT/DICT, STAND)

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
STAN	Imprime os dados de entrada depois da padronização.

Parâmetros apenas para a entrada de matriz

DISSIMILARITIES=**ABSOLUTE**/SIGN

Para INPUT=CORR, especifica como a matriz de dissimilaridade deve ser computada.

ABSO	Considera os valores absolutos dos coeficientes de correlação como medidas de similaridade.
SIGN	Usa os coeficientes de correlação com seus sinais.

MDMATRIX=n

Trata os elementos da matriz iguais a n como dados perdidos.
Default: Todas as variáveis são válidas.

PRINT=MATRIX
Imprime a matriz de entrada.

Parâmetros para ambos os tipos de entrada

VARS=(lista de variáveis)
As variáveis a serem usadas na análise.
Não há default.

ANALYSIS=PAM/FANNY/CLARA/AGNES/DIANA/MONA
Especifica o tipo de análise a ser executada.

PAM	Partição ao redor de medoides.
FANN	Partição com agrupamento difuso.
CLAR	Partição ao redor de medoides (mesmo que PAM), mas para datasets com pelo menos 100 casos. CLUSFIND amostrará os casos e escolherá a amostra mais representativa. Cinco amostras de 40+2*CMAX casos são retiradas (ver o parâmetro CMAX abaixo). Apenas para dados de entrada brutos.
AGNE	Agrupamento hierárquico aglomerativo.
DIAN	Agrupamento hierárquico divisivo.
MONA	Agrupamento monotético de dados consistindo de variáveis binárias. Requer pelo menos 3 variáveis. Apenas para dados de entrada brutos.

Não há default.

CMIN=**2**/n
Para PAM e FANNY. O número mínimo de clusters para testar.

CMAX=n
Para PAM e FANNY, o número máximo de clusters para testar.
Para CLARA, o número exato de clusters para testar.
Default: O maior número entre 20 e o valor especificado por CMIN.

PRINT=(DISSIMILARITIES, GRAPH, TRACE, VNAMES)

DISS	Imprime a matriz de dissimilaridade.
GRAP	Imprime a representação gráfica dos resultados.
TRAC	Imprime cada passo do split binário quando MONA é especificado.
VNAM	Para entrada de matriz, imprime os primeiros 3 dos 8 caracteres dos nomes das variáveis ao invés dos números das variáveis como identificação de objetos.

22.8 Restrições

1. O número máximo de casos que podem ser usados em uma análise (exceto CLARA) é 100.
2. O número mínimo de casos requisitados pela análise de CLARA é 100.
3. O número máximo de objetos em uma matriz de entrada é 100.
4. Apenas 3 caracteres da variável de ID são utilizados nos resultados.

22.9 Exemplos

Exemplo 1. Agrupamento os primeiros 100 casos em 5 grupos usando 6 variáveis quantitativas V11-V16; valores de variáveis são padronizados e distância euclidiana é usada nos cálculos; agrupamento é feito com partição ao redor de medoides; a impressão de gráficos é requerida; casos são identificados pela variável V2.

```
$RUN CLUSFIND
$FILES
```

```
PRINT   = CLUS1.LST
DICTIN  = MY.DIC                        arquivo Dicionário de entrada
DATAIN  = MY.DAT                        arquivo Dados de entrada
$SETUP
PAM ANALYSIS USING RAW DATA AS INPUT
BADD=MD1  VARS=(V11-V16)  STAND  IDVAR=V2  CMIN=5  CMAX=5  PRINT=GRAP
```

Exemplo 2. Agrupamento hierárquico aglomerativo de 30 cidades; a matriz de entrada contém distâncias entre cidades e as cidades são numeradas de 1 a 30; impressão de gráficos é requerida; os nomes das cidades são usados nos resultados.

```
$RUN CLUSFIND
$FILES
PRINT   = CLUS2.LST
FT09    = TOWNS.MAT                     arquivo Matriz de entrada
$SETUP
AGNES ANALYSIS USING MATRIX OF DISTANCES AS INPUT
$COMMENT  ACTUAL DISTANCES WERE DIVIDED BY 10,000 TO BE IN THE INTERVAL 0-1
INPUT=DISS  VARS=(V1-V30)  ANAL=AGNES  PRINT=(GRAP,VNAMES)
```

Capítulo 23

Análise de Configuração (CONFIG)

23.1 Descrição Geral

CONFIG executa análise em um único entrada de configuração espacial na forma de uma matriz retangular do IDAMS (produzida, por exemplo, por MDSCAL). Ela possui a capacidade de centrar, normatizar, rotazar, translar dimensões, computar distâncias entre pontos e computar produtos escalares.

Cada linha de uma matriz de configuração oferece as coordenadas de um ponto de configuração. Portanto, o número de linhas é igual ao número de pontos (variáveis), enquanto o número de colunas é igual ao número de dimensões.

CONFIG pode propiciar saída que permite ao usuário comparar mais facilmente as configurações que originalmente possuiam orientações dissimilares. Ela pode também ser utilizada para executar análises mais aprofundadas na configuração. Rotação, por exemplo, pode fazer com que a configuração seja mais facilmente interpretada.

23.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Seleção de um subconjunto de casos não é aplicável e um filtro não está disponível. Nem também há uma opção dentro do CONFIG para subdividir a configuração de entrada. Uma opção de seleção de uma matriz de um arquivo contendo múltiplas matrizes está disponível dentro de CONFIG (ver o parâmetro DSEQ).

Transformando dados. Uso de declarações de Recode não é aplicável no CONFIG.

Ponderando dados. Uso de variáveis de ponderação não é aplicável.

Tratamento de dados perdidos. CONFIG não reconhece dados perdidos na configuração de entrada. Ordinariamente isso não apresenta nenhum problema, as configurações estão usualmente completas.

23.3 Resultados

Dicionário de matriz de entrada. (Condicional: somente se a matriz de entrada contiver um dicionário. Ver o parâmetro MATRIX). Registros de dicionário de entrada de variável com números correspondentes usados na plotagem (labels de plotagem).

Configuração de entrada. Uma cópia impressa da configuração de entrada.

Configuração centrada. (Opcional: ver o parâmetro PRINT). Se PRINT=ALL ou PRINT=CENT é especificado e a configuração de entrada já está centrada, a mensagem "Configuração de entrada está centrada" é impressa.

Configuração normalizada. (Opcional: ver o parâmetro PRINT). Se PRINT=ALL ou PRINT=NORM é

especificado e a configuração de entrada já está normalizada, a mensagem "Configuração está normalizada" é impressa.

Solução com eixos principais. (Opcional: ver o parâmetro PRINT). As linhas da matriz são os pontos e as colunas os eixos principais. Os elementos da matriz são as projeções dos pontos nos eixos.

Produtos escalares. (Opcional: ver o parâmetro PRINT). A metade inferior esquerda da matriz é impressa. Cada elemento da matriz é o produto escalar para um par de pontos (variáveis).

Distâncias entre pontos. (Opcional: ver o parâmetro PRINT). A metade inferior esquerda da matriz é impressa. Cada elemento da matriz é a distância entre um par de pontos (variáveis). A diagonal, sempre com elementos iguais a zero, é impressa.

Configuração transformada. (Opcional: ver o parâmetro de especificação de transformação PRINT). A configuração transformada é impressa depois da rotação/translação.

Plotagem da configuração transformada. (Opcional: ver o parâmetro de especifcação de transformação PRINT). A configuração transformada é plotada 2 eixos de cada vez depois da rotação/translação. Os pontos são numerados.

Histórico de rotação Varimax. (Opcional: ver o parâmetro PRINT). Um vetor é impresso, o qual contém a variância da matriz de configuração antes de cada ciclo de iteração. Isso é seguido pela matriz de configuração depois da rotação para maximizar o critério de varimax normal. Ela terá o mesmo número de linhas e colunas da matriz de configuração de entrada.

Configuração classificada. (Opcional: ver o parâmetro PRINT). Cada coluna da matriz de configuração, depois de ser ordenada, é impressa horizontalmente ao longo da página.

Plotagens de vetores. (Opcional: ver o parâmetro PRINT). A configuração final é impressa dois eixos de cada vez. Os pontos são numerados usando as labels de plot para variáveis como impressas no dicionário de configuração de entrada.

23.4 Matriz de Configuração de Saída

A configuração final pode ser escrita em um arquivo (ver o parâmetro WRITE). Ela é processada como uma matriz retangular do IDAMS. Ver o capítulo "Dados no IDAMS" para uma descrição de matrizes IDAMS. Registros de identificação de variáveis só serão processados se tais registros estão incluídos no arquivo de configuração (ver o parâmetro MATRIX). O formato para os elementos da matriz é 10F7.3. Os registros contendo os elementos da matriz são identificados por CFG nas colunas 73-75 e um número de seqüência nas colunas 76-80. As dimensões da matriz serão as mesmas dimensões da matriz de entrada.

23.5 Matriz de Distância de Saída

A matriz de distância entre pontos pode ser escrita em um arquivo (ver o parâmetro WRITE). Isso é processado na forma de uma matriz quadrada do IDAMS com registros dummy fornecidos para as médias e desvios-padrões esperados em tal matriz. Registros de identificação de variáveis são produzidos somente se eles são incluídos no arquivo de configuração de entrada (ver o parâmetro MATRIX). O formato dos elementos da matriz é 10F7.3. Os registros contendo os elementos da matriz são identificados por CFG nas colunas 73-75 e um número de seqüência nas colunas 76-80.

23.6 Matriz de Configuração de Entrada

A matriz de entrada deve ser na forma de uma matriz retangular do IDAMS, com, ou sem, registros de identificação de variáveis (ver o parâmetro MATRIX). Ver o capítulo "Dados em IDAMS" para uma descrição desse formato.

Matrizes de configuração obtidas do programa MDSCAL podem ser utilizadas como entrada diretamente em CONFIG.

A matriz de entrada de n(linhas) e m(colunas) deve conter as coordenadas dos n pontos para m dimensões. Não devem haver dados perdidos na matriz de entrada.

Mais de uma configuração pode estar em um arquivo sendo usado como entrada em CONFIG. Aquela a ser analisada é selecionada usando o parâmetro DSEQ.

23.7 Estrutura de Setup

```
$RUN CONFIG

$FILES
     Especificações de arquivo

$SETUP
     1. Título
     2. Parâmetros
     3. Especificações de transformação (condicional)

$MATRIX (condicional)
     Matriz


Arquivos:
FT02       configuração de saída e/ou matriz de distância
FT09       configuração de entrada (omitir se $MATRIX é usado)
PRINT      resultados (default  IDAMS.LST)
```

23.8 Declarações de Controle de Programa

Referir-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle do programa, itens 1-3 abaixo.

1. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  CONFIG EXECUTED AFTER MDSCAL
   ```

2. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  PRINT=(CENT,SORT,DIST) TRANS
   ```

   MATRIX=**STANDARD**/NONSTANDARD
   - STAN Registros de identificação de variável são inclusos na matriz de configuração de entrada.
   - NONS Registros de identificação de variável não são inclusos.

   DSEQ=**1**/n
   - O número da seqüência no arquivo de entrada da configuração que deve ser analisada.

   WRITE=(CONFIG,DISTANCES)
   - CONF Remete a configuração final para outro arquivo.
   - DIST Remete a matriz de distâncias entre pontos para outro arquivo.

   TRANSFORM
   - Especificações de transformação serão supridas.

PRINT=(CENTER, NORMALIZE, PRINAXIS, SCALARS, DISTANCES, VARIMAX, SORTED, PLOT, ALL)

CENT	Muda a origem para o centróide do espaço.
NORM	Altera o tamanho do espaço, tal que a soma dos quadrados dos elementos da matriz seja igual ao número de variáveis.
PRIN	Procura pelos eixos principais.
SCAL	Matriz de produtos escalares.
DIST	Matriz de distâncias entre pontos.
VARI	Rotação ortogonal (varimax) (depois da transformação, se houver).
SORT	Configuração classificada (depois da transformação, se houver).
PLOT	Plota a configuração final.
ALL	Imprime CENT, NORM, PRIN, SCAL, DIST, VARI, SORT, PLOT.

Default: A configuração de entrada é impressa.

Nota. Opções de análise são executadas na configuração de entrada na seqüência especificada acima, a despeito da ordem na qual elas são especificadas com o parâmetro PRINT. Transformações, se houver, são executadas logo antes da rotação ortogonal da configuração. Depois de cada operação, os resultados são impressos. Os efeitos das opções de análise são cumulativos. Se a configuração final é impressa e/ou salva, isso é feito depois de todas as análises terem sido feitas.

3. **Especificações de transformação.** (Condicional: se TRANSFORM foi especificado, use parâmetros como especificados abaixo). Tantas transformações quantas desejadas podem ser especificadas; cada uma deve começar em uma nova linha.

   Se o usuário especifica o ângulo de rotação (DEGREES) e duas dimensões (DIMENSION), a rotação é executada. Se uma constante (ADD) e uma dimensão (DIMENSION) são especificadas, a translação é executada.

```
Exemplo:  DEGR=45, DIME=(5,8) PRINT=PLOT
```

   PRINT=(CONFIG, PLOT)

CONF	Imprime a configuração translada e rotacionada (automático para configurações com 2 dimensões e para a configuração final).
PLOT	Plot a configuração translada ou rotacionada.

   Nota: Não haverá saída impresso para a transformação se PRINT não é especificado. Ele deve ser especificado para cada transformação.

   Parâmetros de rotação

   DIMENSION=(n, m)
   As duas dimensões a serem rotacionadas (apenas rotação pareadas).

   DEGREES=n
   Ângulos de rotação em graus (apenas rotação ortogonal).

   Parâmetros de translação

   DIMENSION=n
   A dimensão a ser translada.

   ADD=n
   Valor a ser adicionado a cada coordenada para a dimensão especificada (pode ser negativa e possuir casas decimais).

23.9 Restrição

O tamanho máximo da matriz de configuração de entrada é 60 linhas por 10 colunas.

23.10 Exemplos

Exemplo 1. Rotação e transformação de uma matriz de configuração previamente criada pelo programa MDSCAL; a configuração final é escrita em um arquivo e, então, impressa; as dimensões 1 e 2 são rotacionadas 60 graus; a dimensão 1 é transformada pela adição de 6.

```
$RUN CONFIG
$FILES
PRINT  = CONF1.LST
FT02   = CONFIG.MAT         arquivo de saída para matriz de configuração
FT09   = MDS.MAT            matriz de configuração de entrada
$SETUP
CONFIGURATION ANALYSIS
PRINT=(PLOT,VARI)  TRAN  WRITE=CONF
DEGR=60  DIME=(1,2)  PRINT=PLOT
ADD=6  DIME=1  PRINT=PLOT
```

Exemplo 2. Computação da matriz de produtos escalares e a matriz de distâncias entre pontos para a 4ª configuração do arquivo de entrada; nenhum plot será requerido.

```
$RUN CONFIG
$FILES
PRINT  = CONF2.LST
FT02   = SCAL.MAT           arquivo de saída para produtos escalares e distâncias
FT09   = MDS.MAT            matriz de configuração de entrada
$SETUP
CONFIGURATION ANALYSIS
PRINT=(SCAL,DIST)  DSEQ=4
```

Capítulo 24

Análise Discriminante (DISCRAN)

24.1 Descrição Geral

A tarefa da análise discriminante é encontrar a melhor função discriminante linear de um conjunto de variáveis que reproduza, tanto quanto possível, um agrupamento a priori de casos considerados.

Um procedimento em passos é utilizado nesse programa, i.e. em cada passo a variável mais poderosa é introduzida na função discriminante. A função critério para selecionar a próxima variável depende do número de grupos especificados (o número de grupos varia de 2 a 20). No caso de dois grupos a distância de Mahalanobis é utilizada. Quando o número de variáveis é maior do que dois, então o critério de seleção de variáveis é o traço do produto da matriz de covariância para as variáveis envolvidas e a matriz de covariância interclasse em um passo particular. Essa é a generalização da matriz de Mahalanobis definida para dois grupos.

Além de executar os principais passos da análise discriminante em uma **amostra básica**, há duas possibilidades opcionais: checagem da potência da função discriminante com a ajuda de uma **amostra de teste**, na qual a designação de grupo dos casos é conhecida (como na amostra básica), mas esses casos não são usados na análise, e classificação dos casos com a ajuda de funções discriminantes fornecidas pela análise em uma **amostra anônima** onde a designação de grupo de casos é desconhecida, ou pelo menos não utilizada.

24.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. Um subseting a mais é possível com o uso de variáveis de amostra e de grupo. Variáveis de análise são selecionadas com o parâmetro VARS.

Transformando dados. Declarações de Recode podem ser usadas.

Ponderando dados. Uma variável pode ser utilizada para ponderar os dados; essa variável de ponderação pode possuir valores inteiros ou decimais. Quando o valor da variável de ponderação para um caso é zero, negativo, perdido ou não-numérico, então o caso é sempre pulado; o número de casos pulados é impresso.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser utilizados para checar a existência de dados perdidos. Casos com dados perdidos na variável de amostra, variável de grupo e/ou variáveis de análise podem ser opcionalmente excluídas da análise.

24.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, apenas para variáveis usadas na execução.

Número de casos nas amostras. O número de casos nas amostras básica, de teste e anônima de acordo com parâmetros definidores da amostra.

Número revisado de casos nas amostras. O número de casos nas amostras básica, de teste e anônima revisados de acordo com os parâmetros definidores de amostra e grupo. Note que as figuras revisadas podem ser menores do que as não revisadas para as amostras básica e de teste se os grupos definidos não cobrem completamente as amostras.

Amostra básica. (Opcional: ver o parâmetro PRINT). As variáveis de identificação e de análise dos casos na amostra básica são impressas por grupos, enquanto os grupos são separados entre si por uma linha de asteriscos.

Amostra teste. Como na amostra básica.

Amostra anônima. Como na amostra básica, exceto que não há grupos.

Estatísticas univariadas. Para cada variável usada na análise, o programa imprime a média dos grupos e desvios-padrões, como também a média total.

Resultados do procedimento passo-a-passo (para cada passo)

Número do passo. O número de seqüência do passo.

Variáveis introduzidas. A lista de variáveis retidas nesse passo.

Função linear discriminante. (Condicional: somente se 2 grupos são especificados). O termo constante e os coeficientes da função linear discriminante correspondem a variáveis já introduzidas.

Tabela de classificação para amostra básica. Tabela de freqüência bivariada mostrando a redistribuição de casos entre os grupos originais e os grupos onde eles são alocados na base da função discriminante, seguida pela percentagem dos casos corretamente classificados.

Tabela de classificação para amostra de teste. Como na amostra básica.

Lista de designação de caso. (Opcional: ver o parâmetro PRINT). Os casos das três amostras são impressas aqui com a identificação de caso, alocação de caso, e valor de função discriminante (para 2 grupos) ou distâncias para cada grupo (para mais de 2 grupos).

Resultados da análise de fator discriminante. (Condicional: somente se mais de 2 grupos especificados). Potência discriminante geral e potência discriminante dos três primeiros fatores, seguidas pelos valores dos fatores discriminantes para médias de grupo. Em adição, uma representação gráfica de casos e médias no espaço dos dois primeiros fatores são também dadas.

24.4 Dataset de Saída

Um dataset com a designação final dos grupos de casos pode ser requisitado. É produzido na forma de arquivo de dados descrito por um dicionário do IDAMS (ver o parâmetro WRITE e o capítulo “Dados em IDAMS”).

Ele contém, na ordem que segue:

- as variáveis transferidas,
- o código dos grupos originais como renumerados por DISCRAN (“Original group” - grupo original),
- o código de grupos designados para casos no final (“Assigned group” - grupo designado),
- o “Sample type” - tipo de amostra (1=básica, 2=de teste, 3=anônima) e,
- para análise como mais de 2 grupos originais, os valores dos dois primeiros fatores discriminantes (“Factor-1”, “Factor-2”).

As variáveis são renumeradas começando de um.

O código dos grupos originais é ajustado para o primeiro código de dados perdidos (999.9999) para casos em amostra anônima; fatores são ajustados para o primeiro código de dados perdidos (999.9999) para casos nas amostras de teste e anônima.

Nota: A variável descrita em IDVAR não é produzida automaticamente e, portanto, variáveis de ID devem ser incluídas na lista de variáveis de transferência.

24.5 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Três tipos de amostras podem ser especificadas no arquivo de entrada:

- amostra básica,
- amostra de teste, e
- amostra anônima.

A análise é baseada na amostra básica. A amostra de teste é usada para testar a função discriminante, enquanto os casos da amostra anônima são simplesmente classificados usando as funções discriminantes.

As amostras são definidas por uma "variável de amostra". A amostra básica não pode estar vazia. Os grupos a serem separados pelas funções discriminantes devem ser definidos por uma "variável de grupo". Essa variável define uma classificação a priori dos casos da amostra básica e de teste.

Todas as variáveis usadas para análise devem ser numéricas; eles podem ter valores inteiros ou decimais. A variável de ID de caso e variáveis a serem transferidas podem ser alfabéticas.

24.6 Estrutura de Setup

```
$RUN DISCRAN

$FILES
     Especificações de arquivo

$RECODE (optional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omit se $DATA é usado)
DICTyyyy   dicionário de saída se WRITE=DATA é especificado
DATAyyyy   dados de saída se WRITE=DATA é especificado
PRINT      resultados (default  IDAMS.LST)
```

24.7 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle do programa, itens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V3=6 OR V11=99
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  DISCRIMINANT ANALYSIS ON AGRICULTURAL SURVEY
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  MDHA=SAMPVAR  IDVAR=V4  SAVAR=R5  BASA=(1,5) VARS=(V12-V15)
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

   MAXCASES=n
   O número máximo de casos (depois da filtragem) a serem usados do arquivo de entrada.
   Default: Todos os casos serão utilizados.

   VARS=(lista de variáveis)
   Lista de variáveis-R ou -V a ser utilizada na análise.
   Não há default.

   MDVALUES=**BOTH**/MD1/MD2/NONE
   Quais valores de dados perdidos devem ser usados para as variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".

   MDHANDLING=(SAMPVAR, GROUPVAR, ANALVARS)
   Escolha de tratamento de dados perdidos.
   SAMP Casos com dados perdidos na variável da amostra são excluídos da análise.
   GROU Casos de amostras básica e de teste com dados perdidos na variável de grupo são excluídos da análise.
   ANAL Casos com dados perdidos nas variáveis de análise são excluídas da análise.
   Default: Casos com dados perdidos são incluídos.

   WEIGHT=número de variável
   O número da variável de ponderação se o dado deve ser ponderado.

   IDVAR=número de variável
   Variável de identificação de caso para os dados e/ou listagem de designação de casos.
   Default: "DISC" é usado como identificador para todos os casos.

   STEPMAX=n
   Número máximo de passos a serem executados. Ele deve ser menor ou igual ao número de variáveis de análise.
   Default: Número de variáveis de análise.

MEMORY=**20000**/n
Memória necessária para execução do programa.

WRITE=DATA
Cria um dataset do IDAMS contendo variáveis transferidas, variáveis de designação de casos, tipo de amostra e valores dos fatores discriminantes, se houver.

OUTFILE=**OUT**/yyyy
Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
Default ddnames: DICTOUT, DATAOUT.

TRANSVARS=(lista de variáveis)
Variáveis (até 99) a serem transferidas para o dataset de saída.

PRINT=(CDICT/DICT, OUTCDICT/OUTDICT, DATA, GROUP)

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
OUTC	Imprime o dicionário de saída com registros-C, se houver.
OUTD	Imprime o dicionário de saída sem registros-C.
DATA	Imprime os dados com as designações de grupo originais dos casos.
GROU	Imprime, para cada caso, a designação de grupo baseada na função discriminante.

Especificação de amostra

Esse parâmetros são opcionais. Se eles não são especificados, todos os casos do arquivo de entrada são interpretados como sendo de uma amostra básica. Amostras de teste e anônima, se elas existirem, devem sempre ser explicitamente definidas. A interseção dos pares de amostras devem ser vazias. Contudo, elas não precisam cobrir o arquivo de dados de entrada completo. Um único valor ou intervalo de valores pode ser usado para selecionar casos que pertecem às amostras correspondentes.

m1 = valor de variável de amostra
ou
m1 <= valor de variável de amostra < m2

onde m1 e m2 podem ser valores inteiros ou decimais.

SAVAR=número de variável
A variável usando para definição de amostra. Variável-R ou -V pode ser usada.

BASA=(m1, m2)
Condicional: define a amostra básica. Deve ser fornecida se SAVAR for especificado.

TESA=(m1, m2)
Condicional e opcional: se SAVAR for especificado. Define a amostra de teste.

ANSA=(m1, m2)
Condicional e opcional: se SAVAR for especificado. Define a amostra anônima.

Classificação da amostra básica

Esses parâmetros definem os grupos a priori no procedimento de análise discriminante. Todos os grupos devem ser definidos explicitamente e a suas interseções em pares devem ser vazias. Contudo, elas não precisam cobrir toda a amostra básica.

GRVAR=número de variável
A variável usada para a definição de grupo. Variável-V ou -R pode ser utilizada.
Não há default.

GR01=(m1, m2)
Define o primeiro grupo na amostra básica.

GR02=(m1, m2)
Define segundo grupo na amostra básica.

GRnn=(m1, m2)
Define o n-ésimo grupo na amostra básica (nn <= 20).

Nota. Pelo menos dois grupos devem ser especificados.

24.8 Restrições

1. Número máximo de grupos a priori é 20.
2. A mesma variável não pode ser usada duas vezes.
3. A largura máxima de campo de variável de ID de caso é 4.
4. Número máximo de variáveis a serem transferidas é 99.
5. Variáveis-R não podem ser transferidas.
6. Se uma variável a ser transferidas é alfabética com largura > 4, apenas os primeiros quatro caracteres são usados.

24.9 Exemplos

Exemplo 1. Análise discriminante em todos os casos juntos; casos são identificados por V1; 5 passos de análise são requisitados; grupos a priori são definidos pela variável V111 que inclui categorias 1-6.

```
$RUN DISCRAN
$FILES
PRINT  = DISC1.LST
DICTIN = MY.DIC                  arquivo Dicionário de entrada
DATAIN = MY.DAT                  arquivo Dados de entrada
$SETUP
CANONICAL LINEAR DISCRIMINANT ANALYSIS
PRINT=(DATA,GROUP)  IDVAR=V1  STEP=5  VARS=(V101-V105)  -
   GVAR=V111  GR01=(1,3)  GR02=(3,5)  GR03=(5,7)
```

Exemplo 2. Repetir a análise descrita no Exemplo 1 usando um subconjunto dos respondentes que possuem o valor de 1 em V5 como a amostra básica e testar os resultados dos respondentes que possuem o valor de 2 em V5.

```
$RUN DISCRAN
$FILES
     como no Exemplo 1
$SETUP
CANONICAL LINEAR DISCRIMINANT ANALYSIS USING BASIC AND TEST SAMPLES
PRINT=(DATA,GROUP)  IDVAR=V1  STEP=5  VARS=(V101-V105)  -
   SAVAR=V5  BASA=1  TESA=2  -
   GVAR=V111  GR01=(1,3)  GR02=(3,5)  GR03=(5,7)
```

Capítulo 25

Funções de Distribuição e de Lorenz (QUANTILE)

25.1 Descrição Geral

QUANTILE gera funções de distribuição, funções de Lorenz, e coeficientes de Gini para variáveis individuais, e executa teste de Kolmogorov-Smirnov entre duas variáveis ou entre duas amostras.

25.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. Além disso, cada análise pode ser executada em um subconjunto adicional com o uso de um parâmetro de filtro. Variáveis a serem analisadas são especificadas com o parâmetro VAR.

Transformando dados. Declarações de Recode podem ser usadas.

Ponderando dados. Uma variável pode ser uasada para ponderar os dados de entrada; essa variável de ponderação pode conter valores inteiros que não sejam maiores que 32.767. Note que pesos com valores decimais são arredondados para o inteiro mais próximo. Quando o valor da variável de ponderação para um caso é zero, negativo, perdido, não-numérico ou excede o máximo, então o caso é "pulado"; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser utilizados para a checagem de dados perdidos. Casos contendo um valor de dados perdidos em uma variável de análise são eliminados dessa análise.

25.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, somente para variáveis usadas na execução.

Resultados para cada análise.

Função de distribuição: mínimo, máximo, e pontos de quebra de subintervalo.
Função de Lorenz (opcional): mínimo, máximo, e pontos de quebra de subintervalo, e coeficiente de Gini.
Curva de Lorenz (opcional): plotada em decis.
Estatísticas do teste de Kolmogorov-Smirnov (opcional).

25.4 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis que são referenciadas (exceto o filtro principal) devem ser numéricas; elas podem ter valores inteiros ou decimais.

25.5 Estrutura de Setup

```
$RUN QUANTILE

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de subconjunto (opcional)
     5. QUANTILE
     6. Especificações de análise (repetida como requisitada)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados

Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default  IDAMS.LST)
```

25.6 Declarações de Controle de Programa

Referir-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, ítens 1-3 e 6 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V5=1
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  MAKING DECILES
   ```

3. **Parâmetros** (mandatório). Para selecionar opções do programa.

   ```
   Exemplo:  MDVAL=MD1, PRINT=DICT
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 carcateres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2
Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
O número máximo de casos (depois da filtragem) a serem utilizados do arquivo de entrada.
Default: Todos os casos serão usados.

MDVALUES=**BOTH**/MD1/MD2/NONE
Quais valores de dados perdidos são utilizados para as variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS". Casos com dados perdidos na análise são eliminados dessa análise.

PRINT=CDICT/DICT

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.

4. **Especificações do subconjunto** (opcional). Essas declarações permitem seleção de um subconjunto de casos para uma análise em particular.

```
Exemplo:  FEMALE   INCLUDE V6=2
```

Regras para codificação

Protótipo: nome declaração

nome
Nome do subconjunto. 1-8 caracteres alfanuméricos iniciando-se com uma letra. Esse nome deve corresponder exatamente ao nome usado em especificações de análise subseqüentes. Espaços em branco intercalados não serão permitidos. É recomendável que todos os nomes sejam justificados a esquerda.

declaração
Definição de subconjunto que segue a sintax da declaração de filtro padrão do IDAMS.

5. **QUANTILE.** A palavra QUANTILE nessa linha sinaliza que a especificação de análise seguirá. Ela deve ser incluída (para separar as especificações de subconjunto das especificações de análise) e deve aparecer apenas uma vez.

6. **Especificações de análise.** As regras de codificação são as mesmas, como para os parâmetros. Cada especificação de análise deve se iniciar em uma nova linha.

```
Exemplos:  VAR=R10  N=5   PRINT=CLORENZ
           VAR=V25  N=10  FILTER=MALE       ANALID=M
           VAR=V25  N=10  FILTER=FEMALE  KS=M
```

VAR=número de variável
Variável a ser analisada.
Não há default.

WEIGHT=número de variável
O número da variável de ponderação se os dados devem ser ponderados. Ponderação de dados não é permitida para o teste de Kolmogorov-Smirnov.

N=**20**/n
Número de subintervalos. Se n<2 ou n>100, um aviso é impresso e o valor de default é usado.

FILTER=xxxxxxxx
: Somente casos que satisfaçam a condição definida na especificação de subconjunto com nome xxxxxxxx será utilizada para essa análise. Inclua o nome entre aspas simples se ele contiver caracteres não-alfanuméricos. Letras maiúsculas devem ser usadas para fazer o match do nome na especificação de subconjunto. Nome esse que será automaticamente convertido para maiúsculas.

ANALID='label'
: Uma label para essa análise, de maneira que ela pode ser referenciada para executar o teste de Kolmogorov-Smirnov. Deve ser incluso entre aspas simples se contiver caracteres não-alfanuméricos.

KS='label'
: Label é uma label designada para uma análise prévia através do parâmetro ANALID e define a variável e/ou amostra com a qual essa análise deve ser comparada usando o teste de Kolmogorov-Smirnov. Deve ser incluso entre aspas simples se contiver caracteres não-alfanuméricos.

PRINT=(FLORENZ, CLORENZ)

FLOR	Imprime a função de Lorenz e coeficiente de Gini.
CLOR	Imprime a curva de Lorenz plotada em decis. (A função de Lorenz é também impressa).

Nota: Se KS é especificado, o parâmetro PRINT é ignorado.

25.7 Restrições

1. O número máximo de variáveis a serem utilizadas (análise + ponderação + filtro local) é 50.
2. O número máximo de casos que podem ser analisados é 5000.
3. O número mínimo de subintervalos é 2; máximo é 100.
4. O número máximo de especificações de subconjuntos é 25.
5. Se estiver utilizando o teste de Kolmogorov-Smirnov, o número máximo de casos é 2500.
6. A função de Lorenz e o teste de Kolmogorov-Smirnov não podem ser requisitados para uma mesma análise.
7. O valores de pontos de quebra são sempre impressos com três casas decimais. Variáveis com mais de três casas decimais são truncadas em três espaços ao serem impressas.

25.8 Exemplo

Geração de uma função de distribuição, função de Lorenz e coeficientes de Gini para a variável V67; análises separadas são executadas no dado todo e, então, em dois subconjuntos; o teste de Kolmogorov-Smirnov é executado para testar a diferença das distribuições da variável V67 em dois subconjuntos dos dados.

```
$RUN QUANTILE
$FILES
PRINT  = QUANT.LST
DICTIN = MY.DIC                arquivo Dicionário de entrada
DATAIN = MY.DAT                arquivo Dados de entrada
$SETUP
COMPARISON OF AGE DISTRIBUTIONS FOR FEMALE AND MALE
*                          (valores default tomados para todos os parâmetros)
FEMALE   INCLUDE V12=1
MALE     INCLUDE V12=2
QUANTILE
VAR=V67  N=15  PRINT=(FLOR,CLOR)
VAR=V67  N=15  PRINT=(FLOR,CLOR)  FILT=FEMALE  ANALID=F
VAR=V67  N=15  PRINT=(FLOR,CLOR)  FILT=MALE
VAR=V67  N=15                     FILT=MALE    KS=F
```

Capítulo 26

Análise Fatorial (FACTOR)

26.1 Descrição Geral

FACTOR cobre um conjunto de análises de componente principal e análise de correspondências possuindo especificações em comum. Ele fornece a possibilidade de executar, com apenas uma leitura da análise fatorial de correspondências dos dados, produtos escalares, produtos normalizados escalares, covariâncias e correlações.

Para cada análise o programa constrói uma matriz representando as relações entre as variáveis e computa seus valores e vetores próprios. Ele, então, calcula os fatores de "caso" e de "variável" dando para cada "caso" e "variável" suas ordenadas, suas qualidades de representação e suas contribuições para os fatores. Uma representação gráfica dos fatores com opções ordinárias ou simplicio-fatoriais podem também ser impressas.

As **variáveis /casos ativos** (principais) são as variáveis/casos em cuja base o procedimento de decomposição fatorial é executado, i.e. elas são usadas na computação da matriz de relações. Pode-se também procurar por uma representação de outras variáveis/casos no espaço de fatores correspondendo às variáveis ativas. Tais variáveis/casos (possuindo nenhuma influência nos fatores) são chamadas **variáveis /casos passivos** (suplementares).

Fala-se a respeito de **representação ordinária** (de variáveis/casos) se os valores (escores de fatores) provenientes diretamente da análise são usados na reprsentação gráfica. Contudo, para um melhor entendimento da relação entre variáveis e casos, outra representação simultânea, a **representação simplicio-fatorial**, é possível.

26.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto dos dados de entrada. Variáveis são selecionadas com os parâmetros PVARS e SVARS.

Transformando dados. Declarações de Recode podem ser usadas.

Ponderando dados. Uma variável pode ser usada para poderação dos dados de entrada; essa variável de ponderação pode assumir valores inteiros ou decimais. Quando o valor da variável de ponderação é zero, negativo, perdido ou não-numérico, então o caso será sempre "pulado"; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser usados para checar a existência de dados perdidos. Existem duas maneira de manusear dados perdidos:

- casos com dados perdidos em variáveis ativas são excluídos da análise,
- casos com dados perdidos em variáveis ativas e/ou passivas são excluídos da análise.

26.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, somente para variáveis usadas na execução.

Estatísticas de resumo. (Opcional: ver o parâmetro PRINT). Número da variável, nome da variável, novo número da variável (renumerada a partir de 1), valores máximo e mínimo, média, desvio-padrão, coeficiente de variação, total, variância, assimetria, curtose e número pondrado de casos válidos para cada variável. Note que desvio-padrão e variância são estimativas baseadas nos dados ponderados.

Dados de entrada. (Opcional: ver o parâmetro PRINT). Grupos de 16 variáveis com, em cada linha: o número correspondente de casos, o total para variáveis ativas e os valores de todas as variáveis, precedido pelo total para as colunas (calculado somente para os casos ativos). Valores são impressos com um ponto decimal explícito e com uma casa decimal. Se mais de 7 caracteres são requisitados para a impressão de um valor, ele será substituído por asteriscos.

Matriz de relações (matriz núcleo). (Opcional: ver o parâmetro PRINT). A matriz (depois da multiplicação por dez elevado a n'ésima potência como indicado na linha impressa antes da matriz), o valor do traço e a tabela de valores e vetores próprios.

Histogramas de valores próprios. O histograma com os percentuais e percentuais cumulativos de cada contribuição do valor próprio para a inércia total. Os traços no histograma mostram o critério de Kaiser para a análise de correlação.

Dicionários dos arquivos de dados de saída. (Opcional: ver o parâmetro PRINT). O dicionário pertencendo aos fatores de "caso" seguido por aquele dos fatores de "variável".

Tabela(s) de fatores. Dependendo das opções escolhidas, haverá: uma tabela (ou para fatores de "caso" ou fatores de "variável"), ou duas tabelas (para ambos fatores de "caso" e "variável", nessa ordem). De acordo com a opção de impressão escolhida, essas tabelas conterão apenas os casos (variáveis) ativos, apenas os passivos, ou ambos.

Tabela de fatores de "caso". Ela dá, linha por linha:

valor de ID do caso,
informação relevante para todos os fatores tomados juntos, i.e. a qualidade de representação do caso no espaço definido pelos fatores, o peso do caso e a "inércia" do caso,
informação para cada fator, i.e. a ordenada do caso, o quadrado do cosseno do ângulo entre o caso e o fator, e a contribuição do caso para o fator.

Tabela de fatores de "variável". Ela dá, linha por linha, informação similar para as variáveis.

Plotagem de dispersão. (Opcional: ver o parâmetro PLOTS). A primeira linha dá o número de fatores representados ao longo do eixo horizontal com seus valores próprios e seus alcances min-max. A segunda linha dá a mesma informação relacionada ao eixo vertical. Juntamente com o título de execução, o número de casos/variáveis (i.e. pontos) que são representados é dado. No lado direito de cada gráfico são impressos:

número de pontos que não podem ser impressos para aquela ordenada (pontos parcialmente coincidentes),
número de pontos que não foram passíveis de representação,
número da página.

Fatores rotacionados. (Opcional: ver o parâmetro ROTATION). A variância calculada para cada matriz de fatores em cada iteração da rotação (usando o método VARIMAX) é impressa, seguida pela comunidade de variáveis antes e depois da rotação, terminando com a tabela de fatores rotacionados.

Mensagem de finalização. No final de cada análise uma mensagem de finalização é impressa com o tipo de análise executada.

26.4 Dataset(s) de Saída

Dois arquivos Dados, cada um com um dicionário IDAMS associado podem opcionalmente ser construídos. No dataset de fatores de "caso", os registros correspondem aos casos (tanto ativos quanto passivos), as colunas correspondem às variáveis (incluindo a identificação de caso e variáveis transferidas) e fatores. No dataset de fatores de "variável", os registros correspondem a variáveis de análise, enquanto as colunas contém as identificações de variável (números de variável originais) e fatores.

Variáveis de saída são numeradas seqüencialmente começando de 1 e elas possuem as seguintes características:

- *Variíveis de identificação de caso (ID) e variáveis transferidas:* Variáveis-V possuem as mesmas características dos seus entradas equivalentes, variáveis Recode são produzidas com WIDTH=9 e DEC=2.
- *Variáveis de fator computadas:*

Nome	especificado por FNAME
Largura de campo	7
No. de decimais	5
MD1 e MD2	9999999

26.5 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis usadas para análise devem ser numéricas; eles podem ter valores inteiros ou decimais. Elas devem ser dicotômicas ou medidas em uma escala intervalar. A variável de ID de caso e variáveis a serem transferidas podem ser alfabéticas. Há dois tipos de variáveis de análise, ativas e passivas. Além disso, uma variável identificando o caso deve existir. Outras variáveis podem ser selecionadas para transferência para o arquivo de dados de saída dos fatores de "caso". Um mais casos no final do arquivo de dados de entrada podem se especificados como casos passivos.

Para análise de correspondência, dois tipos de dados são apropriados: a) variáveis dicotômicas de um arquivo de dados brutos ou b) uma tabela de contingência descrita por um dicionário e usado como entrada do jeito de um dataset do IDAMS.

26.6 Estrutura de Setup

```
$RUN FACTOR

$FILES
     Especificações de arquivo

$RECODE (optional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de plotagem definidas por usuário (condicional)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados

Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída para fatores de caso
DATAyyyy   dados de saída para fatores de caso
DICTzzzz   dicionário de saída para fatores de variáveis
DATAzzzz   dados de saída para fators de variáveis
PRINT      resultados (default  IDAMS.LST)
```

26.7 Declarações de Controle de Programa

Referir-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle do programa, itens 1-4 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  EXCLUDE V10=99 OR V11=99
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  AGRICULTURAL SURVEY 1984
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  ANAL=(CRSP,SSPRO) TRANS=(V16,V20) IDVAR=V1  -
                  PVARS=(V31-V35)
   ```

   INFILE=**IN**/xxxx
   Um sufixo de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
O número máximo de casos (depois da filtragem) a ser usado do arquivo de entrada.
Default: Todos os casos serão utilizados.

MDVALUES=**BOTH**/MD1/MD2/NONE
Quais valores de dados perdidos devem ser utilizados para variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".

MDHANDLING=**PRINCIPAL**/ALL
- PRIN Casos com dados perdidos nas variáveis ativas são excluídos da análise, enquanto casos com dados perdidos nas variáveis passivas são incluídos. Fatores de variáveis passivas são baseados somente em dados válidos.
- ALL Todos os casos com dados perdidos são excluídos.

ANALYSIS=(**CRSP**/NOCRSP, SSPRO, NSSPRO, COVA, CORR)
Escolha de análises.
- CRSP Análise fatorial de correspondências.
- SSPR Análise fatorial de produtos escalares.
- NSSP Análise fatorial de produtos escalares normados.
- COVA Análise fatorial de covariâncias.
- CORR Análise fatorial de correlações.

PVARS=(lista de variáveis)
Lista de variáveis-V ou -R a serem usadas como variáveis ativas.
Não há default.

SVARS=(lista de variáveis)
Lista de variáveis-V ou -R a serem usadas como variáveis passivas.

WEIGHT=número de variável
O número da variável de ponderação se os dados devem ser ponderados.

NSCASES=**0**/n
Número de casos passivos. Nota: Esses casos não são incluídos na computação das estatísticas, matrizes e fatores; eles são os últimos "n" no arquivo de dados.

IDVAR=número de variável
Variável de identificação de caso para pontos na plotagem e para casos no arquivo de saída.
Não há default.

KAISER/NFACT=n/VMIN=n
Critérios para determinação do número de fatores.
- KAIS Critério de Kaiser - número de raízes maiores que 1.
- NFAC Número de fatores desejados.
- VMIN O percentual mínimo de variância a ser explicado pelos fatores tomados todos juntos. Não digite o decimal, e.g. "VMIN=95".

ROTATION=**KAISER**/UDEF/NOROTATION
Especifica a rotação VARIMAX dos fatores de "variável". Somente para análise de correlação.
- KAIS Número de fatores a serem rotacionados é definido de acordo com o critério KAISER.
- UDEF Número de fatores a serem rotacionados é especificado pelo usuário (ver o parâmetro NROT).

NROT=**1**/n
Número de fatores a serem rotacionados (se ROTATION=UDEF especificado).

WRITE=(OBSERV, VARS)
Controla os arquivos de saída de fatores de "caso" e "variável". Se mais de uma análise é requisitada no parâmetro ANALYSIS, esses arquivos serão apenas para a primeira a ser especificada.
OBSE Cria um arquivo contendo fatores de "caso".
VARS Cria um arquivo contendo fatores de "variável".

OUTFILE=**OUT**/yyyy
Um sufixo ddname de 1-4 caacteres para os arquivos Dicionário e Dados para fatores de "caso". Default ddnames: DICTOUT, DATAOUT.

OUTVFILE=**OUTV**/zzzz
Um sufixo ddname de 1-4 caacteres para os arquivos Dicionário e Dados para fatores de "variável". Default ddnames: DICTOUTV, DATAOUTV.

TRANSVARS=(lista de variáveis)
Variáveis (até 99) para ser transferida para o arquivo de fator de "caso" de saída.

FNAME=uuuu
Uma string de 1-4 caracteres usada com um prefixo para nomes de variáveis de fatores nos dicionários de saída. Deve vir entre aspas simples se contiver quaisquer caracteres não-alfanuméricos. Fatores possuem nomes uuuuFACT0001, uuuuFACT0002, etc.
Default: Em branco.

PLOTS=**STANDARD**/USER/NOPLOTS
Controla a representação gráfica dos resultados.
STAN Plotagens padrão serão impressas para pares de fatores 1-2, 1-3, 2-3 com opções PAGES=1, OVLP=LIST, NCHAR=4, REPR=COORD, VARPLOT=(PRINCIPAL,SUPPL).
USER Plotagens definidas pelo usuário são desejáveis (ver os parâmetros para as plotagens definidas por usuário abaixo).

PRINT=(CDICT/DICT, OUTCDICTS/OUTDICTS, STATS, DATA, MATRIX, **VFPRINC**/NOVFPRINC, VFSUPPL, OFPRINC, OFSUPPL)
CDIC Imprime o dicionário de entrada para as variáveis accessadas com registros-C, se houver.
DICT Imprime o dicionário de entrada sem registros-C.
OUTC Imprime os dicionários de saída com registros-C, se houver.
OUTD Imprime os dicionários de saída sem registros-C
STAT Imprime estatísticas das variáveis ativas e passivas.
DATA Imprime os dados de entrada.
MATR Imprime a matriz de relações (matriz núcleo) e vetores próprios.
VFPR Imprime fatores de "variável" para as variáveis ativas.
VFSU Imprime fatores de "variável" para as variáveis passivas.
OFPR Imprime fatores de "caso" para os casos ativos.
OFSU Imprime fatores de "caso" para os casos passivos.

4. **Especificações de plotagem definidas por usuário** (condicional: se PLOT=USER especificada como parâmetro). Repetir para cada plotagem bidimensional a ser impresa. As regras de codificação são as mesmas dos parãmetros. Cada especificação deve iniciar em uma nova linha.

```
Exemplo:  X=3   Y=10
```

X=número do fator
Número do fator a ser representado no eixo horizontal.

Y=número do fator
Número do fator a ser representado no eixo vertical (ver tambémo parâmetro de plotagem FORMAT=STANDARD).

ANSP=**ALL**/CRSP/SSPRO/NSSPRO/COVA/CORR

Especifica a análise para a qual a plotagem deve ser impressa.

ALL Plotagens para todas as análises especificadas no parâmetro ANALYSIS.

Para o resto, uma plotagem para uma análise somente (as palavras-chave possuem o mesmo significado como no parâmetro ANALYSIS). Essas opções implicam apenas uma plotagem.

OBSPLOT=(PRINCIPAL, SUPPL)

Escolha dos casos a serem representados na(s) plotagen(s).

PRIN Representa casos ativos.

SUPP Representa casos passivos.

VARPLOT=(**PRINCIPAL**/NOPRINCIPAL, SUPPL)

Escolha das variáveis a serem representadas na(s) plotagen(s).

PRIN Representa variáveis ativas.

SUPP Representa variáveis passivas.

REPRESENT=**COORD**/BASVEC/NORMBV

Escolha da representação simultânea dos pontos (variáveis/casos).

COOR Coordenadas como indicado na tabela de fatores.

BASV Representa vetores básicos.

NORM Representa vetores básicos usando uma norma especial para representação simplicio-fatorial.

OVLP=**FIRST**/LIST/DEN

Opção relativa a representação de pontos quase coincidentes.

FIRS Imprime o número/ID de caso da variável para o primeiro ponto apenas.

LIST Dá uma lista vertical dos pontos possuindo a mesma abscissa no gráfico até um outro ponto ser encontrado (o número/ID de caso da variável é então perdido).

DEN Imprime a densidade (número de pontos quase coincidentes). Imprime para um ponto ".", para dois (quase conincidente) pontos ":", para trêz "3", etc, para 9 pontos "9", para mais de 9 pontos "*". NCHAR=2 deve ser especificado se essa opção for selecionada.

NCHAR=**4**/n

Número de dígitos/caracteres usados para a identificação das variáveis/casos na(s) plotagem(ens) (1 a 4 caracteres).

PAGES=**1**/n

Número de páginas por plotagem.

FORMAT=**STANDARD**/NONSTANDARD

Define o tamanho da moldura da plotagem.

STAN Usa uma moldura de 21 x 30 cm para a plotagem mostrando o fator com a maior extensão no eixo horizontal e usando diferentes escalas para os dois eixos.

NONS A moldura não será padronizada no sentido do descrito acima. O tamanho da plotagem é definido por PAGES=n, e significado dos eixos X e Y.

26.8 Restrições

1. Número máximo de variáveis de análise é 80.
2. Uma (e somente uma) variável de identificação deve ser especificada.
3. Número máximo de variáveis a serem transferidas é 99.
4. Número máximo de variáveis de entrada incluindo aquelas nas declarações de filtro e Recode é 100.

5. Número máximo de 24 plotagens definidas por usuário.

6. Se a variável de ID ou uma variável a ser transferida é alfabética com largura > 4, apenas os primeiros quatro caracteres são usados.

7. Para os parâmetros o seguinte deve ser atendido:

   max(D1,D2,D3) < 5000

   onde
   D1 = NPV * NPV + 10 * NV
   D2 = NV * (NF + 6) + NPV * NIF
   D3 = NV + NF + NIF + 3 * NP

   e NV, NPV, NF, NIF, NP significam o número total de variáveis de análise, número de variáveis ativas, número de fatores a serem computados, número de fatores a serem ignorados, número máximo de pontos a ser repesentado nas plotagens, respectivamente.

26.9 Exemplos

Exemplo 1. Análise fatorial de correlações; análises são baseadas em 20 variáveis e 7 fatores são requisitados; o número de fatores a ser rotacionado é definido de acordo com o critério de Kaiser; estatísticas, matriz de correlação e vetores próprios serão impressos, seguidos pelos fatores de variável e plotagens padrão; os fatores não serão mantidos no arquivo.

```
$RUN FACTOR
$FILES
PRINT  = FACT1.LST
DICTIN = A.DIC                    arquivo Dicionário de entrada
DATAIN = A.DAT                    arquivo Dados de entrada
$SETUP
FACTOR ANALYSIS OF CORRELATIONS
ANAL=(NOCRSP,CORR)  ROTA=KAISER  NFACT=7  IDVAR=V1  PRINT=(STATS,MATRIX)  -
    PVARS=(V12-V16,V101-V115)
```

Exemplo 2. Análise fatorial de produtos escalares baseado em 10 variáveis; 2 variáveis passivas, V5 e V7, devem ser representadas em plotagens; plotagens são definidas pelo usuário, pois somente o primeiro ponto dos pontos quase coincidentes é requerido; os critérios de Kaiser devem ser usados para determinar o número de fatores; ambos, fatores de caso e de variável, serão escritos em arquivos.

```
$RUN FACTOR
$FILES
DICTIN   = A.DIC                  arquivo Dicionário de entrada
DATAIN   = A.DAT                  arquivo Dados de entrada
DICTOUT  = CASEF.DIC              arquivo Dicionário para fatores de caso
DATAOUT  = CASEF.DAT              arquivo Dados para fatores de caso
DICTOUTV = VARF.DIC               arquivo Dicionário para fatores de variável
DATAOUTV = VARF.DAT               arquivo Dados para fatores de variável
$SETUP
FACTOR ANALYSIS OF SCALAR PRODUCTS
ANAL=(NOCRSP,SSPR)  IDVAR=V1  WRITE=(OBSERV,VARS) PRINT=STATS  PLOT=USER  -
   PVARS=(V112-V116,V201-V205)  SVARS=(V5,V7)
X=1  Y=2  VARP=(PRINCIPAL,SUPPL)
X=1  Y=3  VARP=(PRINCIPAL,SUPPL)
X=2  Y=3  VARP=(PRINCIPAL,SUPPL)
```

Exemplo 3. Análise de correspondência usando uma tabela de contingência descrita por um dicionário e inserida como um datasen no arquivo Setup para ser executado; número de fatores é definido pelos critério de Kaiser; matriz de relações será impressa, seguida pelos fatores de variável e caso, e pelas plotagens de variáveis e casos.

```
$RUN FACTOR
$FILES
PRINT   = FACT3.LST
$SETUP
CORRESPONDENCE ANALYSIS ON CONTINGENCY TABLE
BADD=MD1  IDVAR=V8  PLOTS=USER  PRINT=(MATRIX,OFPRINC) PVARS=(V31-V33)
$DICT
$PRINT
   3   8  33   1   1
T   8 Scientific degree           1   20
C   8         81     Professor
C   8         82     Ass.Prof.
C   8         83     Doctor
C   8         84     M.Sc
C   8         85     Licence
C   8         86     Other
T  31 Head                        4   20
T  32 Scientifc                   7   20
T  33 Technician                 10   20
$DATA
$PRINT
81  5  0  0
82  1  3  0
83  0 17 01
84  0 28 04
85  0  0 01
86  0  0 17
```

Capítulo 27

Regressão Linear (REGRESSN)

27.1 Descrição Geral

REGRESSN oferece uma capacidade de regressão múltipla geral designada tanto para regressão linear padrão quanto stepwise. Várias análises de regressão, usando diferentes parâmetros e variáveis, podem ser feitas com uma execução.

Termo constante. Se a entrada consiste de dados brutos, o usuário pode requisitar que a equação não contenha termo constante (ver o parâmetro de regressão CONSTANT=0). Nesse caso, uma matriz baseada na matriz de produto cruzado é analisada, ao invés da matriz de correlação. Isso muda a inclinação da linha ajustada e pode afetar substancialmente os resultados. Na regressão stepwise, as variáveis podem entrar na equação em uma ordem diferente daquela quando um termo constante é estimado. Se uma matriz de correlação é a entrada, a equação de regressão sempre incluirá um termo constante.

Uso de variáveis categóricas como variáveis independentes. Uma opção está disponível para criar um conjunto de variáveis dummy (dicotômicas) para variáveis categóricas especificadas (ver o parâmetro CATE). Elas podem ser usadas como variáveis independentes na análise de regressão.

F-ratio para uma variável a entrar na equação. Numa regressão stepwise, variáveis são adicionadas, por vez, na equação de regressão até que a equação se torne satisfatória. A cada passo, a variável com a maior correlação parcial com a variável dependente é selecionada. Um valor de teste-F parcial é então calculado para a variável e esse valor é comparado com um valor crítico fornecido pelo usuário. Assim que o F parcial para a próxima variável a ser introduzida se torna menor que o valor crítico, a análise é terminada.

F-ratio para uma variável a ser removida da equação. Uma variável que tinha sido a melhor variável a entrar em um estágio anterior de uma regressão stepwise pode, em um estágio posterior, não ser mais a melhor por causa da relação entre ela e outras variáveis agora na regressão. Para detectar isso, o valor parcial de F para cada variável na regressão é, a cada passo do cálculo, computado com um valor crítico fornecido pelo usuário. Qualquer variável cujo valor-F se encontra abaixo do valor crítico é removida do modelo.

Regressão stepwise. Se uma regressão stepwise é requisitada, o programa determina que variáveis ou quais conjuntos de variáveis dummy dentre o conjunto de variáveis independentes especificadas serão realmente usadas para a regressão, e em qual ordem elas serão introduzidas, começando com as variáveis forçadas e continuando com as outras variáveis e conjuntos de variáveis dummy, uma a uma. Depois de cada passo, o algoritmo seleciona, das variáveis preditoras remanescentes, a variável ou conjunto de variáveis dummy que produz a maior redução na variância dos resíduos (não-explicada) da variável dependente, a menos que a sua contribuição ao F-ratio total para a regressão permaneça abaixo de um nível especificado. Similarmente, o algoritmo avalia depois de cada passo se a contribuição de qualquer variável ou conjunto de variáveis dummy já inclusa cai abaixo de um nível especificado, em cujo caso ela é eliminada da regressão.

Regressão stepwise descendente. Como a regressão stepwise, exceto que o algoritmo inicia com todas as variáveis independentes e então elimina variáveis ou conjuntos de variáveis de uma maneira stepwise. A cada passo o algoritmo seleciona, das variáveis preditoras inclusas remanescentes, a variável ou conjunto de variáveis dummy que produz a menor redução na variância explicada da variável dependente, a não ser que isso exceda um nível especificado. Similarmente, o algoritmo avalia a cada passo, se a contribuição de qualquer variável ou conjunto de variáveis dummy previamente eliminada da regressão aumentou acima de

um nível especificado, em cujo caso ela é adicionada de volta a regressão.

Gerando um dataset de resíduos. Com dados brutos, resíduos podem ser computados e obtidos como um arquivo de dados descrito por um dicionário do IDAMS. Ver a seção "Dataset(s) de Resíduos de Saída" para detalhes do conteúdo. Note que um dataset separado de resíduos é gerado para cada equação. Também, como REGRESSN não possui nenhuma facilidade para transferir variáveis de interesse específicas em uma análise residual dos dados brutos de entrada para um dataset de resíduos, pode ser necessário o uso do programa MERGE para criar um dataset contendo todas as variáveis desejadas. Uma variável de ID de caso do dataset de entrada é produzida para o dataset de resíduos para tornar o emparelhamento possível.

Gerando uma matriz de correlação. Se dados brutos são usados como entrada, o program computa coeficientes de correlação que podem ser produzidos no formato de uma matriz quadrada do IDAMS e usado para análises posteriores. Correlações de REGRESSN incluem todas as variáveis ao longo das equações de regressão e são baseadas em casos que possuem dados válidos em todas as variáveis na matriz. Portanto, correlações usualmente irão diferir das correlações obtidas com a execução do programa PEARSON com a opção MDHANDLING=PAIR. Quando a eliminação de dados perdidos em REGRESSN deixa um tamanho de amostra aceitavelmente grande, REGRESSN é uma alternativa a PEARSON para gerar uma matriz de correlação (ver o parágrafo "Tratamento de dados perdidos").

27.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Se dados brutos são usados como entrada, o filtro padrão está disponível para selecionar um subconjunto de dados dos dados de entrada. Se uma matriz de correlação é utilizada como entrada no programa, seleção de caso não é aplicável. As variáveis para a equação de regressão são especificadas nos parâmetros de regressão DEPVAR e VARS.

Transformando dados. Se dados brutos são usados como entrada, declarações de Recode podem ser usadas.

Ponderando dados. Se dados butos são usados como entrada, uma variável pode ser usada para ponderar os dados de entrada; essa variável de ponderação pode possuir valores inteiros ou decimais. O programa força o somatório dos pesos a ser igual ao número de casos de entrada. Quando o valor da variável de ponderação para um caso é zero, negativo, perdido, ou não-numérico, então o caso é sempre "pulado"; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos.

1. **Entrada.** Se dados brutos são usados como entrada, o parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser usados para checar a existência de dados perdidos. Casos onde dados perdidos ocorrem em qualquer variável de regressão em qualquer análise são deletados (a deleção de dados perdidos "por casos"). Uma opção (ver o parâmetro MDHANDLING) permite ao usuário especificar o número máximo de casos de dados perdidos que podem ser tolerados antes da execução ser terminada. Atenção: Se análises múltiplas são executadas em uma execução de REGRESSN, uma matriz única de correlação é computada para todas as variáveis usadas em diferentes análises. Por causa do método "por casos" de deleção de casos com dados perdidos, o número de casos utilizados e, portanto, a estatística de regressão produzida pode ser diferente caso a análise seja executada separadamente.

   Se uma matriz é a entrada, casos com dados perdidos devem ter sido acomodados quando a matriz foi criada. Se uma célula da matriz de entrada possui código de dados perdidos (i.e. 99.999) qualquer análise envolvendo aquela célula será evitada.

2. **Resíduos de saída.** Se resíduos são requisitados, valores previstos e resíduos são computados para todos os casos que passem o filtro (opcional). Se um caso possui dados perdidos em qualquer variável requisitada para essas computações, códigos de dados perdidos de saída são gerados.

3. **Matriz de correlação de saída.** O algoritmo REGRESSN para manusear dados perdidos em dados brutos de entrada não pode resultar em entradas de dados perdidos na matriz de correlação.

27.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, apenas para variáveis usadas na execução.

Estatísticas univariadas. (Dados brutos como entrada apenas). A soma, média, desvio-padrão, coeficiente de variação, máximo, e mínimo são impressos para todas as variáveis dependentes e independentes usadas.

Matriz de soma total de quadrados e produtos cruzados. (Dados brutos como entrada apenas. Opcional: ver o parâmetro PRINT).

Matriz de soma de quadrados residuais e produtos cruzados. (Dados brutos como entrada apenas. Opcional: ver o parâmetro PRINT).

Matriz de correlação total. (Opcional: ver o parâmetro PRINT).

Matriz de correlação parcial. (Opcional para cada regressão: ver o parâmetro de regressão PARTIALS). O elemento ij e a correlação parcial entre a variável i e a variável j, mantendo constante as variáveis especificadas na lista de variáveis PARTIALS.

Matriz inversa. (Opcional para cada regressão: ver o parâmetro de regressão PRINT).

Estatísticas de resumo de análise. As seguintes estatísticas para cada regressão ou para cada passo de uma regressão stepwise:

erro padrão de estimação,
F-ratio,
coeficiente de correlação múltipla (ajustado e não-ajustado),
fração da variância explicada (ajustado e não-ajustado),
determinante da matriz de correlação,
graus de liberdade dos resíduos,
termo constante.

Estatísticas de análise para preditores. As seguintes estatísticas são impressas para cada regressão ou cada passo de uma regressão stepwise:

coeficiente B (coeficiente de regressão parcial não-padronizado),
erro-padrão (sigma) de B,
coeficiente beta (coeficiente de regressão parcial padronizado),
erro-padrão (sigma) de B,
R quadrado parcial e marginal,
t-ratio,
quociente de covariância,
valores de R quadrado marginal para todos os preditores e quocintes T para todos os conjuntos de variáveis dummy (para regressão stepwise).

Dicionário de resíduos de saída. (Para dados brutos como entrada apenas. Opcional: ver o parâmetro de regressão WRITE).

Dados de resíduos de saída. (Para dados brutos como entrada apenas. Opcional: ver o parâmetro de regressão PRINT). Se há menos de 1000 casos, valores calculados, valores observados e resíduos (diferenças) podem ser listados em ordem ascendente de valor de resíduo. Qualquer número de casos pode ser listado em ordem de seqüência de entrada do caso. A estatística de Durbin-Watson para os resíduos associados será impressa para resíduos listados em ordem de seqüência de entrada do caso.

27.4 Matriz de Correlação de Saída

A matriz de correlação pode ser computada (ver o parâmetro WRITE). Ela é escrita na forma de uma matriz quadrada do IDAMS (ver o capítulo "Dados em IDAMS"). O formato é 6F11.7 para as correlações e 4E15.7 para as médias e desvios-padrões. Além disso, infomação de labels é escrita nas colunas 73-80 dos registros da seguinte maneira:

registro definidor de matriz	N=nnnnn
registros de correlação	REG xxx
registros de médias	MEAN xxx
registros de desvio-padrão	SDEV xxx

(nnnnn é o tamanho da amostra de REGRESSN. O xxx é o número de seqüência começando com 1 para o primeiro registro de correlação e incrementado de um para cada registro sucessivo através do registro do último desvio-padrão).

Os elementos da matriz são r's de Pearson. Eles, como também as médias e desvios-padrões, são baseados nos casos que possuem dados válidos em todas as variáveis especificadas em qualquer das listas de variáveis de regressão. As correlações são para todos os pares de variáveis de todas as listas de variáveis de análise tomadas juntas.

27.5 Dataset(s) de Resíduos de Saída

Para cada análise, um dataset de resíduos pode ser requisitado (ver o parâmetro de regressão WRITE). Isso é produzido na forma de um arquivo Dados descrito por um dicionário do IDAMS. Ele contém quatro ou cinco variáveis por caso, se os dados são ou não ponderados: uma variável de ID, uma variável dependente, uma variável dependente predita (calculada), um resíduo, e um peso, se houver. Casos são produzidos na ordem dos casos de entrada. As características do dataset são as seguintes:

	No. de variável	Nome	Largura de campo	No. de decimais	Código de MD1
(variável de ID)	1	mesmo do entrada	*	0	mesmo do entrada
(variável dependente)	2	mesmo do entrada	*	**	mesmo do entrada
(variável predita)	3	Predicted value	7	***	9999999
(resíduo)	4	Residual	7	***	9999999
(peso-se ponderado)	5	mesmo do entrada	*	**	mesmo do entrada

*	transferido do dicionário de entrada para variáveis V ou 7 para variáveis R
**	transferido do dicionário de entrada para variáveis V ou 2 para variáveis R
***	6 mais no. de decimais para variável dependente menos parâmetro de largura da variável dependente; se isso for negativo, então 0.

Se o valor calculado ou resíduo exceder a largura de campo alocada, será substituído pelo código MD1.

27.6 Dataset de Entrada

O dataset de dados brutos de entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis usadas para análise devem ser numéricas; elas podem ter valores inteiros ou decimais. A variável de ID de caso pode ser alfabética.

27.7 Matriz de Correlação de Entrada

Isso é uma matriz quadrada do IDAMS. Uma matriz de correlação gerada por PEARSON ou por uma REGRESSN prévia é uma matriz de entrada apropriada para REGRESSN.

O dicionário da matriz de entrada deve conter números e nomes de variáveis. A matriz deve conter correlações, médias e desvios-padrões. Ambas, média e desvios-padrões, são utilizados.

27.8 Estrutura de Setup

```
$RUN REGRESSN

$FILES
     Especificações de arquivo

$RECODE (opcional com dados brutos como entrada; indisponível com uma matriz)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Definição de variáveis dummy (condicional)
     5. Especificações de regressão (repetido como requisitada)

$DICT (condicional)
     Dicionário para entrada de dados brutos

$DATA (conditional)
     Dados para entrada de dados brutos

$MATRIX (condicional)
     Matriz para entrada de matriz de correlação


Arquivos:
FT02       matriz de correlação de saída
FT09       matriz de correlação de entrada
           (se $MATRIX não é usada e INPUT=MATRIX)
DICTxxxx   dicionário de entrada (se $DICT não é usado e INPUT=RAWDATA)
DATAxxxx   dados de entrada (se $DATA não é usado e INPUT=RAWDATA)
DICTyyyy   dicionário de resíduos de saída)  um conjunto para cada
DATAyyyy   dados de resíduos de saída     )  arquivo de resíduos requisitado
PRINT      resultados (default  IDAMS.LST)
```

27.9 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-3 e 5 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução. Disponível apenas com entada de dados brutos.

   ```
   Exemplo:  INCLUDE V3=5
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  REGRESSION ANALYSIS
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  IDVAR=V1  MDHANDLING=100
   ```

INPUT=**RAWDATA**/MATRIX

RAWD Os dados de entrada estão em uma forma de arquivo Dados descrito por um dicionário do IDAMS.

MATR Os dados de entrada são coeficientes de correlação e estão em uma forma de matriz quadrada do IDAMS.

Parâmetros somente para entrada de dados brutos

INFILE=**IN**/xxxx

Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2

Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n

O número máximo de casos (depois da filtragem) a serem usados do arquivo de entrada.
Default: Todos os casos serão utilizados.

MDVALUES=**BOTH**/MD1/MD2/NONE

Quais valores de dados perdidos devem ser usados para variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".

MDHANDLING=**0**/n

O número de casos de dados perdidos a serem permitidos antes do término da execução. Um caso é contado como perdido se ele possuir um dado perdido em qualquer uma das variáveis nas equações de regressão.

WEIGHT=número de variável

O número da variável de peso se os dados devem ser ponderados.

CATE

Especifica CATE se a definição de uma variável dummy é oferecida.

IDVAR=número de variável

Variável a ser produzida ou impressa como ID de caso se dataset de resíduos são requisitado. As variáveis de ID não devem ser incluídas em qualquer lista de variáveis.

WRITE=MATRIX

Escreva a matriz de correlação computada dos dados de entrada para um arquivo de saída.

PRINT=(CDICT/DICT, XMOM, XPRODUCTS, MATRIX)

CDIC Imprime o dicionário de entrada para variáveis acessadas com registros-C, se houver.

DICT Imprime o dicionário de entrada sem registros-C.

XMOM Imprime a matriz de soma de quadrados residuais e produtos cruzados.

XPRO Imprime a matriz de soma total de quadrados e produtos cruzados.

MATR Imprime a matriz de correlação.

Parâmetros para a entrada da matriz de correlação

CASES=n

Faça CASES igual ao número de casos usados para criar a matriz de entrada. Esse número é usado no cálculo do nível de F.
Não há default; deve ser fornecido quando do entrada da matriz de correlação.

PRINT=MATRIX

Imprime a matriz de correlação.

4. **Definição de variáveis dummy** (condicional: se CATE foi especificado como um parâmetro). O programa REGRESSN pode transformar uma variável categórica em um conjunto de variáveis dummy. Para se ter uma variável tratada como categórica, o usuário deve a) incluir o parâmetro CATE na lista de parâmetros e b) especificar as variáveis a serem consideradas como categóricas e os códigos a serem usados. Cada variável categórica a ser transformada é seguida pelos códigos as serem utilizados entre parênteses. Para cada variável, quaisquer códigos não listados serão excluídos da construção. Nota: A lista de códigos não deve ser exaustiva, i.e. não deve-se listar todos os códigos existentes, caso isso ocorra uma matriz singular resultará.

   ```
   Exemplo:  V100(5,6,1), V101 (1-6)
   ```

   Códigos 5, 6 e 1 da variável 100 será representado na regressão como variáveis dummy, juntamente com códigos 1 até 6 da variável 101.

Uma variável especificada na definição de variáveis dummy, quando usada em lista de variáveis previsoras (VARS), parciais (PARTIALS) ou forçadas (FORCE) para regressão stepwise, se referirá ao conjunto de variáveis dummy criado daquela variável. Em regressões stepwise, os códigos de tal variável serão introduzidos ou excluídos juntos, e R-quadrados marginais e F-ratios serão calculados para todos os códigos das variáveis juntamente, como também para códigos individualmente. Uma variável usada na definição de variáveis dummy pode não ser utilizada como variável dependente.

5. **Especificações de regressão.** As regras de codificação são as mesmas das dos parâmetros. Cada conjunto de parâmetros de regressão deve se iniciar em uma nova linha.

   ```
   Exemplo:  DEPV=V5  METH=STEP  FORCE=(V7) VARS=(V7,V16,V22,V37-V47,R14)
   ```

METHOD=**STANDARD**/STEPWISE/DESCENDING

STAN	Uma regressão padrão será feita.
STEP	Uma regressão stepwise será feita.
DESC	Uma regressão stepwise descendente será feita.

DEPVAR=número de variável
Número da variável dependente.
Não há default.

VARS=(lista de variáveis)
As variáveis independentes a serem utilizadas nessa análise.
Não há default.

PARTIALS=(lista de variáveis)
Computa e imprime uma matriz de correlação parcial com as variáveis especificadas removidas da lista de variáveis independentes.
Default: Nenhuma parcial.

FORCE=(lista de variáveis)
Força as variáveis listadas a entrarem na regressão stepwise (METH=STEP) ou a permanecerem na regressão stepwise descendente (METH=DESC).
Default: Não força.

FINRATIO=**.001**/n
O valor do F-ratio abaixo do qual uma variável não entrará no procedimento stepwise; esse é o F-ratio a ser entrado. O ponto decimal deve ser entrado.

FOUTRATIO=**0.0**/n
O valor do F-ratio acima do qual uma variável permanecerá no procedimento stepwise; esse é o F-ratio a ser removido. O ponto decimal deve ser entrado.

CONSTANT=0
Somente para dados brutos como entrada.
O termo constante é requerido ser igual a zero e nenhum termo constante será estimado.
Default: Um termo constante será estimado.

WRITE=RESIDUALS
Resíduos devem ser escritos como um dataset do IDAMS.

OUTFILE=**OUT**/yyyy
Aplicável somente se WRITE=RESI é especificado.
Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de resíduos de saída. Se produzindo resíduos de mais de 1 análise, o ddname default, OUT, só poderá ser usado uma vez.

PRINT=(STEP, RESIDUALS, ERESIDUALS, INVERSE)

STEP	Aplica-se a regressão stepwise apenas: imprime R-quadrado marginal para todos os previsores em cada passo.
RESI	Imprime os resíduos na ordem de seqüência de caso de entrada e estatística de Durbin-Watson.
ERES	Imprime resíduos, exceto para dados perdidos, em ordem de magnitude de erro, desde que existam menos do que 1000 casos.
INVE	Imprime a matriz de correlação inversa.

27.10 Restrições

1. Com dados brutos como entrada, podem haver 99 ou 100 (dependendo se uma variável de ponderação é utilizada) variáveis diferentes usadas em uma única equação de regressão; o número total de variáveis ao longo de todas as análises, incluindo variáveis de Recode, variável de ponderação e variável de ID, não pode ser maior que 200.
2. Com entrada de uma matriz, a matriz pode ser 200 x 200, e até 100 variáveis podem ser usadas em uma única equação de regressão.
3. FINRATIO deve ser maior ou igual a FOUTRATIO.
4. Resíduos podem ser listados em ordem ascendente de valor do resíduo somente se há menos de 1000 casos.
5. Uma variável especificada em uma definição de variáveis dummy pode não ser usada como variável dependentes.
6. Um máximo de 12 variáveis dummy podem ser definidas de uma variável categórica.
7. Se a variável de ID é alfabética com largura > 4, somente os primeiros quatro caracteres são usados.

27.11 Exemplos

Exemplo 1. Regressão padrão com cinco variáveis independentes usando uma matriz de correlação do IDAMS como entrada.

```
$RUN REGRESSN
$FILES
FT09 = A.MAT                          arquivo Matrix de entrada
SETUP
STANDARD REGRESSION  -  USING MATRIX AS INPUT
INPUT=MATR  CASES=1460
DEPV=V116  VARS=(V18,V36,V55-V57)
```

Exemplo 2. Regressão padrão com seis variáveis independentes e com duas variáveis cada uma com 3 categorias transformadas em 6 variáveis dummy; dados brutos são utilizados com entrada; resíduos devem ser computados e escritos em um conjunto de dados (casos são identificados pela variável V2).

```
$RUN REGRESSN
$FILES
PRINT   = REGR2.LST
DICTIN  = STUDY.DIC                    arquivo Dicionário de entrada
DATAIN  = STUDY.DAT                    arquivo Dados de entrada
DICTOUT = RESID.DIC                    arquivo Dicionário para resíduos
DATAOUT = RESID.DAT                    arquivo Dados para resíduos
$SETUP
STANDARD REGRESSION  -  USING RAW DATA AS INPUT AND WRITING RESIDUALS
MDHANDLING=50  IDVAR=V2  CATE
V5(1,5,6),V6(1-3)
DEPV=V116  WRITE=RESI  VARS=(V5,V6,V8,V13,V75-V78)
```

Exemplo 3. Duas regressões: uma padrão e uma stepwise usando dados brutos como entrada.

```
$RUN REGRESSN
$FILES
DICTIN = STUDY.DIC                     arquivo Dicionário de entrada
DATAIN = STUDY.DAT                     arquivo Dados de entrada
$SETUP
TWO REGRESSIONS
PRINT=(XMOM,XPROD)
DEPV=V10  VARS=(V101-V104,V35)  PRINT=INVERSE
DEPV=V11  METHOD=STEP  PRINT=STEP VARS=(V1,V3,V15-V18,V23-V29)
```

Exemplo 4. Regressão em dois estágios; o primeiro estágio usa as variáveis V2-V6 para estimar valores da variável dependente V122; no segundo estágio, duas variáveis adicionais V12, V23 são usadas para estimar os valores preditos de V122, i.e. V122 com os efeitos de V2-V6 removidos.

Na primeira regressão, valores preditos da variável dependente (V122) são computados e escritos para o arquivo de resíduos (OUTB) como a variável V3. MERGE é então utilizado para fundir essa variável com as variáveis do arquivo original que são requisitadas no segundo estágio. O conjunto de dados produzido de MERGE (um arquivo temporário, portanto não precisa ser definido) conterá 5 variáveis da lista de construção, numeradas V1 a V5 onde A12 e A23 (a serem utilizadas como preditores no segundo estágio) tornan-se V2 e V3, A122, a variável dependente original, se torna V4, e B3, a variável dando valores preditos de V122 se torna V5. Esse arquivo de saída é então usado como entrada na regressão de segundo estágio.

```
$RUN REGRESSN
$FILES
PRINT    = REGR4.LST
DICTIN   = STUDY.DIC                    arquivo Dicionário de entrada
DATAIN   = STUDY.DAT                    arquivo Dados de entrada
DICTOUTB = RESID.DIC                    arquivo Dicionário para resíduos
DATAOUTB = RESID.DAT                    arquivo Dados para resíduos
$SETUP
TWO STAGE REGRESSION  -  FIRST STAGE
MDHANDLING=100  IDVAR=V1
DEPV=V122  WRITE=RESI  OUTF=OUTB  VARS=(V2-V6)
$RUN MERGE
$SETUP
MERGING PREDICTED VALUE (V3 IN RES FILE) INTO DATA FILE
MATCH=INTE  INAF=IN  INBF=OUTB
A1=B1
A1,A12,A23,A122,B3
$RUN REGRESSN
$SETUP
TWO STAGE REGRESSION  -  SECOND STAGE
MDHANDLING=100  INFI=OUT
DEPV=V5  VARS=(V2,V3)
```

Capítulo 28

Escalonamento Multidimensional (MDSCAL)

28.1 Descrição Geral

MDSCAL é um programa de escalonamento multidimensional não-métrico para a análise de similaridades. O programa, que opera em uma matriz de medidas de similaridade ou dissimilaridade, é desenhado para encontrar, para cada dimensionalidade especificada, a melhor representação geométrica dos dados no espaço.

Os usos de escalonamento multidimensional são similares aqueles de análise fatorial, e.g. clusters de variáveis podem ser encontrados, a dimensionalidade dos dados pode ser descoberta e as dimensões podem às vezes ser interpretadas. O programa CONFIG pode ser usado para executar a análise em uma configuração de saída do MDSCAL.

Configuração de entrada. Normalmente uma configuração inicial arbitrária criada é utilizada para começar a computação. O usuário pode, contudo, fornecer uma configuração inicial. Há várias razões possíveis para se fornecer uma configuração inicial. O usuário pode ter razões teóricas para iniciar com uma certa configuração; pode-se desejar executar mais iterações em uma configuração que ainda não está próxima o suficiente de uma configuração ótima; ou, para economizar tempo de processamento, pode-se desejar fornecer uma configuração com maior dimensão como ponto de partida para uma configuração com menos dimensões.

Algoritmo de escalonamento. O programa inicia com uma configuração inicial, gerada arbitrariamente ou fornecida pelo usuário, e produz iterações (usando um procedimento tipo "steepest descent") ao longo de sucessivas configurações de teste, a cada vez comparando a ordem de posição das diferenças inter-pontuais na configuração de teste com a ordem de posição da medida correspondente nos dados. Uma medida de "mal ajuste" (coeficiente de esforço) é computada a cada iteração e a iteração é arranjada novamente, de acordo, para melhorar o ajuste aos dados, até, idealmente, a ordem de posição das distâncias na configuração seja perfeitamente monotônica com a ordem de posição das dissimilaridades dadas pelos dados: o "esforço" será zero. Na prática, a computação do escalonamento pára, em qualquer número de dimensões, por que o esforço alcança um valor suficientemente pequeno (STRMIN), o fator de escala (magnitude) do gradiente alcança um valor suficientemente pequeno (SRGFMN), o esforço vem melhorando muito vagarosamente (SRATIO), ou o número presente máximo de iterações é alcançado (INTERATIONS). O programa é interrompido em qualquer uma das condições que aconteça primeiro. O mesmo procedimento é repetido para a próxima dimensionalidade menor usando os resultados anteriores como configuração inicial, até que um número mínimo de dimensões especificado seja alcançado. Durante a computação, o cosseno do ângulo entre gradientes sucessivos possui um papel importante em vários sentidos; opcionalmente, dois parâmetros de ponderação internos podem ser especificados (ver parâmetros COSAVW e ACSAVW).

Dimensionalidade e métrica. Soluções podem ser obtidas de 2 a 10 dimensões. O usuário controla a dimensionalidade das configurações obtidas especificando o número máximo e mínimo de dimensões desejadas, e a diferença entre a dimensionalidade de sucessivas soluções produzidas (ver os parâmetros DMAX, DMIN, e DDIF). O usuário também especifica, usando o parâmetro R, se a métrica de distância deve ser euclidiana (R=2), o caso usual, ou outra métrica-r de Minkowski.

Esforço. Esforço é uma medida de quão bem a configuração se compatibiliza com os dados. O usuário pode escolher entre duas fórmulas alternadas para computação do coeficiente de esforço: ou o esforço é padronizado pela soma das distâncias quadradas da média (SQDIST) ou o esforço é padronizado pela soma dos desvios quadrados da média (SQDEV). Em muitas situações, as configurações alcançadas pelas duas fórmulas não serão suficientemente diferentes. Maiores valores do esforço resultam da fórmula 2 para um mesmo grau de ajuste.

Empates nos coeficientes de entrada. Há dois métodos alternativos de lidar com empates entre os valores de dados de entrada; as distâncias correspondentes podem ser requeridas a serem iguais (TIES=EQUAL) ou elas podem ser permitidas diferirem (TIES=DIFFER). Quando há poucos empates, faz pouca diferença qual abordagem é utilizada. Quando há muitos empates a abordagem utilizada faz diferença, e o contexto deve ser considerado ao se fazer a escolha.

28.2 Características Padrão do IDAMS

Seleção de casos e variáveis. A filtragem de casos deve ser feita na hora em que a matriz é criada, não em MDSCAL. O parâmetro VARS permite que a computação seja executada em subconjuntos da matriz, ao invés da matriz inteira.

Transformando dados. Uso de declarações de Recode não é aplicável no MDSCAL. Transformações de dados devem ser executadas na hora que a matriz de entrada é criada.

Ponderando dados. Ponderação no sentido usual (ponderando casos para diferentes taxas de amostragem ou diferentes níveis de agregação) deve ser feita antes do uso de MDSCAL; tal ponderação deve ser incorporada na matriz de dados de entrada. Há uma opção de ponderação, de um tipo bem diferente, disponível em MDSCAL (ver o parâmetro INPUT=WEIGHTS). Ele deve ser usado para designar pesos para células da matriz de entrada; o usuário fornece uma matriz de valores que devem ser usados como pesos para os elementos correspondentes na matriz de entrada.

Tratamento de dados perdidos. Dados perdidos para casos individuais devem ser contabilizados no momento que a matriz de dados de entrada é criada, não em MDSCAL. Se, depois que a matriz tenha sido criada, uma entrada na matriz é dado perdido, i.e. contém um código de dados perdidos, há uma possibilidade de processá-la usando MDSCAL: a opção de cutoff do MDSCAL (ver o parâmetro CUTOFF) pode ser usada para excluir da análise valores de dados perdidos se eles forem menores que valores de dados válidos. MDSCAL não possui nenhuma opção para reconhecer código de dados perdidos que sejam números grandes (como 99.99901, o código de dados produzido por PEARSON). Se grandes valores de dados perdidos existirem, eles devem ser transformados em números pequenos. Se uma variável em particular possui várias entradas perdidas, possivelmente ela deve ser descartada da análise.

28.3 Resultados

Matriz de entrada. (Opcional: ver o parâmetro PRINT).

Pesos de entrada. (Opcional: ver o parâmetro PRINT).

Configuração de entrada. Se uma configuração inicial é fornecida, ele é sempre impressa.

História dos cálculos. Para cada solução, o programa imprime um histórico completo das computações, reportando o valor de esforço e os seus parâmetros auxiliares para cada iteração:

Iteração	o número da iteração
Esforço	o valor corrente do esforço
SRAT	o valor corrente do quociente de esforço
SRATAV	a média corrente do quociente de esforço (é uma média exponencialmente ponderada)
CAGRGL	o cosseno do ângulo entre o gradiente corrente e gradiente anterior
COSAV	o valor corrente do cosseno médio do ângulo entre gradientes sucessivos (uma média ponderada)
ACSAV	o valor corrente do valor absoluto médio do cosseno do ângulo entre gradientes sucessivos (uma média ponderada)
SFGR	o comprimento (mais propriamente, o fator de escala) do gradiente
STEP	o tamanho do passo.

Razão para interrupção. Quando a computação é interrompida, a razão é indicada por um dos seguintes avisos: "Foi esforço o mínimo", "Número máximo de iterações usado", "Foi alcançado esforço satisfatório", ou "Foi alcançado esforço zero".

Configuração final. Para cada solução, as coordenadas Cartesianas da configuração final são impressas.

Configuração classificada. (Opcional: ver o parâmetro PRINT). Para cada solução, as projeções dos pontos da configuração final são classificadas separadamente em cada dimensão em ordem ascendente e, então, impressas.

Resumo. Para cada solução, os valores de dados originais são classificados e impressos juntamente com suas distâncias finais correspondentes (DIST) e as distâncias hipotéticas requeridas para um ajuste monotônico perfeito (DHAT).

28.4 Matriz de Configuração de Saída

Ao ser calculada a configuração final para cada dimensionalidade, ela pode ser processada como uma matriz retangular do IDAMS. A configuração é centralizada e normalizada. As linhas representam as variáveis e as colunas as dimensões. Os elementos da matriz são escritos em formato 10F7.3. Registros de dicionário são gerados. Essa matriz pode ser submetida como um entrada de configuração para outra execução de MDSCAL ou ela pode ser entrada para outro programa como CONFIG para análise adicional.

28.5 Matriz de Dados de Entrada

A entrada usual para MDSCAL é uma matriz quadrada do IDAMS (ver o capítulo "Dados em IDAMS"). Essa matriz é a metade superior direita da matriz sem a diagonal e é definida pelo parâmetro INPUT=STANDARD. TABLES e PEARSON geram matrizes adequadas para serem entrada de MDSCAL. Médias e desvios-padrões não são usados, mas registros (dummy) apropriados devem ser fornecidos. MDSCAL aceitará matrizes em outros formatos além do o triangular superior direito sem a diagonal. Contudo, tais matrizes devem conter a porção do dicionário de uma matriz quadrada do IDAMS e deve ter registros contendo pseudo médias e desvios-padrões no final.

Os seguintes parâmetros de INPUT indicam o formato exato da matriz sendo usada como entrada:

STAN	triângulo superior-direito, sem diagonal
STAN, DIAG	triângulo superior-direito, com diagonal
LOWER, DIAG	triângulo inferior-esquerdo, com diagonal
LOWER	triângulo inferior-esquerdo, sem diagonal
SQUARE	matriz quadrada completa com diagonal.

As medidas contidas na matriz de dados podem ser medidas de similaridade (como correlações) ou dissimilaridades. Apesar do entrada para MDSCAL ser usualmente uma matriz de coeficientes de correlação (e.g. a uma matriz de gamas ou uma matriz de r's de Pearson), a matriz de entrada pode conter qualquer medida que faça sentido como uma medida de proximidade. Devido ao fato de que escalonamento não-numérico utilizar somente propriedades ordinais dos dados, nada precisa ser assumido em relação às propriedades quantitativas ou numéricas dos dados. Deve haver, no mínimo, duas vezes mais variáveis do que dimensões.

28.6 Matriz de Ponderação de Entrada

Se uma matriz de ponderação é fornecida, ela deve estar exatamente no mesmo formato da matriz de dados de entrada. O parâmetro INPUT=(STAN/LOWE/SQUA, DIAG) se aplica a matriz de ponderação como também a matriz de dados. O dicionário para a matriz de ponderação deve ser o mesmo da matriz de dados de entrada. Médias e desvios-padrões não são usados, mas linhas "dummy" correspondentes devem ser fornecidas.

Essa matriz contém valores, em correspondência um-para-um com os elementos da matriz de dados, que devem ser usados como pesos para os dados. Esses valores são usados em conjunção com o valor para o parâmetro CUTOFF quando aplicados aos dados. Se um valor de dados é maior que o valor de cutoff, mas o valor correspondente do peso é menor ou igual a zero, uma condição de erro é sinalizada. Do mesmo jeito, se o valor de dados é menor ou igual ao valor de cutoff, e o valor de peso correspondente é maior que zero, uma condição de erro é imposta. Se qualquer uma dessas inconsistências ocorrem, a execução é terminada.

28.7 Matriz de Configuração de Entrada

A matriz de configuração de entrada deve estar no formato de uma matriz retangular do IDAMS. Ver o capítulo "Dados em IDAMS".

Ela propicia uma configuração inicial a ser usada nas computações. As linhas devem representar as variáveis e as colunas, dimensões. Ela é usualmente produzida por uma execução prévia de MDSCAL e é submetida para que uma execução prévia possa começar onde ela foi deixada.

A matriz deve conter pelo menos tantas dimensões quanto o valor dado para o parâmetro DMAX.

Nota: Se a lista de variável (VARS) é especificada, MDSCAL usa as **primeiras** n linhas da configuração de entrada onde n é o número de variáveis na lista, sem checar os números das variáveis.

28.8 Estrutura de Setup

```
$RUN MDSCAL

$FILES
     Especificações de arquivo

$SETUP
     1. Título
     2. Parâmetros

$MATRIX (condicional)
     Matriz de dados
     Matriz de pesos
     Matriz de configuração inicial
(Nota:  Nem totas as matrizes devem ser incluídas aqui; contudo, se
 mais de uma matriz é incluída, elas devem estar na ordem acima).

Arquivos:
FT02    matriz de configuração de saída
FT03    matriz de ponderação de entrada se INPUT=WEIGHTS é especificado
        (omitir se $MATRIX é usado)
FT05    configuração inicial de entrada se INPUT=CONFIG  é especificado
        (omitir se $MATRIX é usado)
FT08    matriz de dados de entrada (omitir se $MATRIX é usado)
PRINT   resultados (default  IDAMS.LST)
```

28.9 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-2 abaixo.

1. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  MDSCAL EXECUTION ON DATASET X4952
   ```

2. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  DMAX=5  ITER=75  WRITE=CONFIG
   ```

INPUT=(**STANDARD**/LOWER/SQUARE, DIAGONAL, WEIGHTS, CONFIG)

STAN	A entrada é uma matriz quadrada do IDAMS, i.e. fora da diagonal, metade superior direita da matriz.
LOWE	A matriz de entrada é uma metade inferior esquerda da matriz.
SQUA	A matriz de entrada é uma matriz completa.
DIAG	A matriz de entrada possui os elementos da diagonal.
WEIG	Uma matriz de valores de pesos está sendo fornecida.
CONF	A matriz de configuração inicial está sendo fornecida.

VARS=(lista de variáveis)

Lista de variáveis na matriz onde a análise está sendo executada.
Default: A matriz de entrada inteira é utilizada.

FILE=(DATA, WEIGHTS, CONFIG)

DATA	A matriz de dados de entrada está em um arquivo.
WEIG	A matriz de ponderação está em um arquivo.
CONF	A matriz de configuração de entrada está em um arquivo.

Default: Todas as matrizes são assumidas seguir um comando $MATRIX na ordem dados, peso, configuração.

COEFF=**SIMILARITIES**/DISSIMILARITIES

SIMI	Altos coeficientes na matriz de dados indicam que pontos são similares ou próximos.
DISS	Altos coeficientes na matriz de dados indicam que pontos são dissimilares ou afastados.

DMAX=**2**/n

O máximo da dimensão: escalonamento inicia com o espaço de dimensão máxima.

DMIN=**2**/n

O mínimo da dimensão: escalonamento procede até alcançar ou passar a dimensão mínima.

DDIF=**1**/n

A diferença de dimensão: escalonamento procede da dimensão máxima até a dimensão mínima em passos de diferença de dimensões.

R=**2.0**/n

Indica que a métrica-r de Minkowski deve ser usada. Qualquer valor >= 1.0 pode ser usado.

R=1.0	Métrica de city-block.
R=2.0	Distância euclidiana ordinária.

CUTOFF=**0.0**/n

Valores de dados menores ou iguais a n são descartados. Se os valores legítimos dos coeficientes de entrada vão de -1.0 a 1.0, CUTOFF=-1.01 deve ser usado.

TIES=**DIFFER**/EQUAL

DIFF Distâncias diferentes correspondendo a valores de dados iguais não contribuem para o coeficiente de esforço e nenhuma tentativa é feita para equalizar essas distâncias.

EQUA Distâncias diferentes correspondendo a valores de dados iguais contribuem para o esforço e há uma tentativa para equalizar essas distâncias.

ITERATIONS=**50**/n

O número máximo de iterações a serem executadas em qualquer número de dimensões. Esse máximo é uma precaução de segurança para controlar o tempo de execução.

STRMIN=**.01**/n

Mínimo esforço. O procedimento de escalonamento terminará se o esforço alcança o valor mínimo.

SFGRMN=**0.0**/n

Valor mínimo do valor do fator de escala do gradiente. O procedimento de escalonamento terminará se o gradiente alcançar esse valor mínimo.

SRATIO=**.999**/n

O quociente de esforço. O procedimento de escalonamento pára se o quociente de esforço entre passos sucessivos alcança n.

ACSAVW=**.66**/n

O fator de ponderação para o valor absoluto médio do cosseno do ângulo entre gradientes sucessivos.

COSAVW=**.66**/n

O fator de ponderação para o cosseno médio do ângulo entre sucessivos gradientes.

STRESS=**SQDIST**/SQDEV

SQDI Computa o esforço utilizando a padronização pela soma das distâncias quadradas.

SQDE Computa o esforço utilizando a padronização pela soma dos desvios quadrados da média.

WRITE=CONFIG

Produz a configuração final de cada solução em um arquivo.

PRINT=(MATRIX, SORTCONF, **LONG**/SHORT)

MATR Imprime a matriz de dados de entrada e a matriz de pesos se uma é fornecida.

SORT Classifica cada dimensão da configuração final e imprime isso.

LONG Imprime matrizes em linhas longas.

SHOR Imprime matrizes em linhas curtas.

28.10 Restrições

1. A capacidade do programa é de 1800 pontos de dados (1800 elementos da matriz de similaridade ou dissimilaridade). Isso é equivalente a um triângulo de uma matriz 60 x 60 ou a uma matriz quadrada de 42 x 42.

2. Variáveis podem ser escalonadas em até 10 dimensões.

3. A matriz de configuração inicial pode ter um máximo de 60 linhas e 10 colunas.

28.11 Exemplo

Geração de uma matriz de configuração de saída; a matriz de dados de entrada está em uma forma padrão do IDAMS e em um arquivo; não há nem matriz de ponderação de entrada, nem matriz de configuração de entrada; 20 iterações são requisitadas; análise deve ser executada em um subconjunto das variáveis.

```
$RUN MDSCAL
$FILES
FT02 = MDS.MAT                  arquivo Matriz de configuração de saída
FT08 = ABC.COR                  arquivo Matriz de dados
$SETUP
MULTIDIMENSIONAL SCALING
ITER=20  WRITE=CONFIG  FILE=DATA  VARS=(V18-V36)
```

Capítulo 29

Análise de Classificação Múltipla (MCA)

29.1 Descrição Geral

MCA examina a relação entre várias variáveis preditoras e uma única variável dependente e determina os efeitos de cada preditor antes e depois do ajustamento por suas inter-correlações com outros preditores na análise. Ele também fornece informação sobre as relações bivariadas e multivariadas entre os preditores e a variável dependente. A técnica MCA pode ser considerada o equivalente a uma análise de regressão múltipla usando variáveis dummy. MCA, contudo, é muitas vezes mais conveniente de usar e interpretar. MCA possui também uma opção para análise de variância univariada.

MCA assume que os efeitos dos preditores são aditivos i.e. que não existem interações entre os preditores. Ele é desenhado para uso com variáveis preditoras medidas em escalas nominal, ordinal, e intervalos. Ele aceita um número desigual de casos nas células formadas pela classificação cruzada dos preditores.

Alternativas ao MCA são REGRESSN e ONEWAY. REGRESSN oferece uma capacidade de regressão múltipla. ONEWAY executa uma análise de variância univariada. A vantagem do MCA em relação REGRESSN é que ele aceita variáveis preditoras em uma forma tão "fraca" quanto escalas nominais, e ela não assume linearidade na regressão. As vantagens em relação a ONEWAY são que em MCA o código máximo para uma variável de controle em uma análise univariada é 2999 (ao invés de 99 em ONEWAY).

Gerando um dataset de resíduos. Resíduos podem ser computados e produzidos como um arquivo Dados descrito por um dicionário do IDAMS. Ver a seção "Dataset(s) de Resíduos de Saída" para detalhes desse conteúdo. A opção não está disponível se somente um preditor é especificado.

Procedimentos iterativos. MCA usa um algoritmo de iteração para aproximar os coeficientes constituindo as soluções para o conjunto de equações normais. O algoritmo de iteração pára quando os coeficientes sendo gerados estão suficientemente precisos. Isso envolve colocar uma tolerância e especificar um teste para determinar quando aquela tolerância foi alcançada (ver parâmetros de análise CRITERION e TEST). Quatro testes de convergência estão disponíveis. Se os coeficientes não convergem dentro dos limites colocados pelo usuário, o programa imprime seus resultados na base da última iteração. O número de iterações úteis depende de uma certa maneira do número de preditores usados na análise e na fração especificada para tolerância. Se há menos do que 10 preditores, usualmente tem-se achado satisfatório especificar 10 como o número máximo de iterações.

Detecção e tratamento de interações. O programa assume que o fenômeno sendo examinado pode ser entendido em termos de um modelo aditivo.

Se, aprioristicamente, variáveis particulares são suspeitas de interagirem, MCA pode ser usado para determinar a extensão da interação da seguinte maneira. Se um preditor é especificado, MCA executa uma análise de variância univariada. Tal análise pode ajudar a detectar e eliminar interações de preditores. O procedimento completo é como se segue (ver também Exemplo 3):

1. Determina um conjunto de preditores suspeitos de interagirem.
2. Forma uma única "variável de combinação" usando esses preditores e a declaração de Recode COMBINE.
3. Executa uma análise de MCA usando os preditores suspeitos para obter o R quadrado ajustado.
4. Executa a análise de MCA com a "variável de combinação" como o controle em uma análise de variância univariada para obter o eta quadrado ajustado, que será maior ou igual ao R quadrado ajustado.
5. Usa a diferença, eta quadrado ajustado - R quadrado ajustado (a fração da variância explicada que é perdida por causa da hipótese de aditividade), como um guia para determinar se o uso de uma variável de combinação no lugar dos preditores originais é justificado.

O teste para interação deve ser baseado na mesma amostra da execução normal de MCA. Se interações são detectadas, então a variável de combinação deve ser usada como variável preditora no lugar das variáveis individuais que interagem.

29.2 Características Padrão do IDAMS

Seleção de casos e variáveis. Casos podem ser excluídos de todas as análises na execução do MCA pelo uso de uma declaração de filtro padrão. Em análise de classificação múltipla, casos podem ser excluídos também por excederem o código máximo do preditor. (Nota: Se uma variável preditora de qualquer análise possui um código fora do intervalo 0-31, o caso contendo o valor é eliminado de todas as análises). Para qualquer análise particular, casos adicionais podem ser excluídos devido às seguintes condições:

- Um caso (referido como outlier) possui um valor de variável dependente que é maior que um número especificado de desvios-padrões da média da variável dependente. Ver parâmetros de análise OUTDISTANCE e OUTLIERS.
- Um caso tem um valor de variável dependente que é maior do que um máximo especificado. Ver o parâmetro de análise DEPVAR.
- Um caso possui um dado perdido para a variável dependente ou variável de ponderação. Ver os parágrafos "Tratamento de dados perdidos" e "Ponderando dados" abaixo.

Transformando dados. Declarações de Recode podem ser usadas.

Ponderando dados. Uma variável pode ser usada para ponderar os dados de entrada; essa variável de ponderação pode conter valores inteiros ou decimais. Quando o valor da variável de ponderação é zero, negativo, perdido ou não-numérico, então o caso é sempre evitado; o número de casos tratados dessa maneira é impresso. Quando dados ponderados são usados, testes de significância estatística devem ser interpretados com cuidado.

Tratamento de dados perdidos. O parâmetro de análise MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser usados para checar a existência de dados perdidos na variável dependente. Casos com dados perdidos na variável dependente são sempre excluídos. Casos com dados perdidos nas variáveis preditoras podem ser excluídos de todas as análises usando o filtro. (Usar o filtro para excluir casos com dados perdidos na variáveis preditoras em classificação múltipla somente é necessário se os códigos de dados perdidos estão no intervalo 0-31; se o valor para qualquer preditor está fora desse intervalo, um caso é automaticamente excluído de todas as análises na execução).

29.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, apenas para variáveis usadas na execução.

Tabela de freqüência ponderada. (Opcional: ver o parâmetro de análise PRINT). Uma matriz N x M é impressa para cada par de preditores onde N=maximum código do preditor de linha e M=maximum código do preditor de coluna. O número total de tabelas é P(P-1)/2 onde P é o número de preditores.

Coeficientes para cada iteração. (Opcional: ver o parâmetro de análise PRINT). Os coeficientes para cada classe para cada preditor.

Estatísticas da variável dependente. Para a variável dependente (Y):
grand média, desvio-padrão e coeficiente de variação,
soma de Y e soma de Y-quadrado,
soma de quadrados total, explicada e residual,
número de casos usados na análise e soma dos pesos.

Estatísticas de preditores para análise de classificação múltipla.

Para cada categoria de cada preditor:
o código da categoria (classe), e label se existir no dicionário,
o número de casos com dados válidos (em forma bruta, ponderada e percentual),
média (não-ajustada e ajustada), desvio-padrão e coeficiente de variação da variável dependente,
desvio não-ajustado da média da categoria em relação a grand média e, coeficiente de ajustamento.

Para cada variável preditora:
eta e eta quadrado (não-ajustado e ajustado),
beta e beta quadrado,
soma dos quadrados não-ajustada e ajustada.

Estatísticas de análise para análise de classificação múltipla. Para todos os preditores combinados:
R-quadrado múltiplo (não-ajustado e ajustado),
fator de ajuste para graus de liberdade,
R múltiplo (ajustado),
listagem dos betas em ordem decrescente dos seus valores.

Estatístivas de análise de variância univariada.

Para cada categoria do preditor:
o código da categoria (classe), e label se existir no dicionário,
o número de casos com dados válidos (em foema bruta, ponderada e percentual),
média, desvio-padrão e coeficiente de variação da variável dependente,
valores da soma e percentual da variável dependente,
soma dos valores da variável dependente ao quadrado.

Para variáveis preditoras:
eta e eta quadrado (não-ajustado e ajustado),
fator de ajuste para graus de liberdade,
soma dos quadrados total, entre médias e dentro de grupos,
valor de F (graus de liberdade é impresso).

Resíduos. (Opcional: ver o parâmetro de análise PRINT). A variável identificadora, valor observado, valor predito, variável residual e de ponderação, se houver, são impressas para casos na ordem do arquivo de entrada.

Estatísticas de resumo dos resíduos. Se os resíduos são requisitados, o programa imprime o número de casos, soma de pesos, e média, variância, assimetria, e curtose da variável de resíduo.

29.4 Dataset(s) de Resíduos de Saída

Para cada análise, resíduos podem opcionalmente ser produzidos em um arquivo Dados descrito por um dicionário do IDAMS. (Ver o parâmetro de análise WRITE=RESIDUALS). Um registro é produzido para cada caso passando um filtro contendo uma variável de ID, um valor observado, um valor calculado, um valor residual para a variável dependente em um valor da variável de ponderação, se houver. As características do dataset são as seguintes:

	No. de variável	Nome	Largura de campo	No. de decimais	Códigos de MD
(variável de ID)	1	mesmo do entrada	*	0	mesmo do entrada
(variável dependente)	2	mesmo do entrada	*	**	mesmo do entrada
(variável predita)	3	Predicted value	7	***	9999999
(resíduo)	4	Residual	7	***	9999999
(peso-se ponderado)	5	mesmo do entrada	*	**	mesmo do entrada

* transferido do dicionário de entrada para variáveis V ou 7 para variáveis R
** transferido do dicionário de entrada para variáveis V ou 2 para variáveis R
*** 6 mais no. de decimais para a variável dependente menos a largura da variável dependent; se for negativo, então 0.

Se o valor observado ou variável de ponderação é perdido ou o caso foi excluído pela checagem de um código máximo ou por um critério de outlier, um registro de resíduo é produzido com todas as variáveis (exceto a variável identificadora) igualadas a MD1.

29.5 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário IDAMS. Todas as variáveis usadas na análise devem ser numéricas; elas podem ter valores inteiros ou decimais, exceto para preditores, que devem ter valores inteiros, entre 0 e 31 para classificação múltipla e até 2999 para análise de variância univariada. A variável de ID de caso pode ser alfabética.

Um grande número de casos é necessário para uma análise de MCA; um boa regra de bolso é que o número total de categorias (i.e. á soma de categorias ao longo de todos os preditores) não deve exceder 10% do tamanho da amostra.

A variável dependente deve ser medida em uma escala intervalar ou ser dicotômica, e não deve ser muito assimétrica. Variáveis preditoras para MCA devem ser categorizadas, preferencialmente com não mais de 6 categorias. Apesar de MCA ser desenhado para manusear preditores correlacionados, nenhum par de preditores deve ser fortemente correlacionado de maneira que haja um overlap perfeito entre suas categorias. (Se houver um overlap perfeito, recodificação para combinar categorias ou filtragem para remover casos ofensivos é necessário).

29.6 Estrutura de Setup

```
$RUN MCA

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de análise (repetida como requisitada)

$DICT (condicional)
     Dictionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de resíduos de saída)  um conjunto para cada
DATAyyyy   dados de resíduos de saída     )  arquivo de dados requisitado
PRINT      resultados (default  IDAMS.LST)
```

29.7 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais detalhadas das declarações de controle de programa, itens 1-4 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V6=2-6
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  TEST RUN FOR MCA
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  *
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dictionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
O número máximo de casos (depois da filtragem) a serem usados do arquivo de entrada.
Default: Todos os casos serão utilizados.

PRINT=CDICT/DICT
CDIC Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT Imprime o dicionário de entrada sem os registros-C.

4. **Especificações de análise.** As regras de codificação são as mesmas dos parâmetros. Cada especificação de análise deve começar em uma nova linha.

```
Exemplo:  PRINT=TABLES, DEPVAR=(V35,98), ITER=100, CONV=(V4-V8)
```

DEPVAR=(número de variável, maxcode)
Número da variável e código máximo para a variável dependente.
Não há default; o número da variável deve ser sempre especificado.
Default para maxcode é 9999999.

CONVARS=(lista de variáveis)
Variáveis a serem usadas como preditores. Se apenas uma variável é dada, uma análise de variância univariada será executada.
Não há default.

MDVALUES=**BOTH**/MD1/MD2/NONE
Quais valores de dados perdidos para a variável dependente devem ser utilizados. Ver o capítulo "O Arquivo Setup do IDAMS".
Nota: Valores de dados perdidos nunca são checados para variáveis preditoras.

WEIGHT=número de variável
O número da variável de ponderação se os dados forem ponderados.

ITERATIONS=**25**/n
O número máximo de iterações. Intervalo: 1-99999.

TEST=**PCTMEAN**/CUTOFF/PCTRATIO/NONE
O teste de convergência desejado.
PCTM Testa se a mudança em todos os coeficientes de uma iteração para a próxima está abaixo de uma fração especificada da grand média.
CUTO Testa se a mudança em todos os coeficientes de uma iteração para a próxima é menor do que um valor especificado.
PCTR Testa se a mudança em todos os coeficientes de uma iteração para a próxima está abaixo de uma fração especificada do quociente entre o desvio-padrão da variável dependente e sua média.
NONE O program continuará processando até o número máximo de iterações ter sido excedida.

CRITERION=**.005**/n
Fornece um valor numérico que é a tolerância do teste de convergência selecionado. Ele vai de 0.0 a 1.0 (Entre o ponto decimal).

OUTLIERS=**INCLUDE**/EXCLUDE
INCL Casos com valores outlying da variável dependente serão contabilizados e incluídos na análise.
EXCL Outliers serão excluídos da análise.

OUTDISTANCE=**5**/n
: Número de desvios-padrões em relação a sua grand média usados para definir um outlier para a variável dependente.

WRITE=RESIDUALS
: Escreva os resíduos em um dataset do IDAMS; aplique o modelo MCA apenas para o subconjunto de dados passando os critérios de dados perdidos, código-máximo, e outlier. Casos onde o modelo MCA não se aplica são incluídos no dataset dos resíduos com todos os valores (exceto o valor da variável identificadora) ajustados para MD1.
Resíduos não podem ser obtidos se apenas uma variável preditora for especificada.

OUTFILE=**OUT**/yyyy
: Um sufixo ddname de 1-4 caracteres para os arquivos Dictionário e Dados de saída dos resíduos. Default ddnames: DICTOUT, DATAOUT.
Nota: Se mais de uma análise requisita o saída de resíduos, os ddnames de default DICTOUT e DATAOUT só podem ser utilizados uma vez.

IDVAR=número de variável
: Número de uma variável de identificação a ser incluída no dataset de resíduos.
Default: Uma variável é criada cujos valores são números indicando a posição seqüencial do caso no arquivo de resíduos.

PRINT=(TABLES, HISTORY, RESIDUALS)

TABL	Imprime as tabulações cruzadas emparelhadas dos preditores.
HIST	Imprime os coeficientes de todas as iterações. Se a opção HIST não é selecionada e se as iterações convergem, somente os coeficientes finais são impressos; se as iterações não convergem, somente os coeficientes das 2 últimas iterações são impressos.
RESI	Imprime resíduos na ordem de seqüência de entrada de caso.

29.8 Restrições

1. O número máximo de variáveis de entrada, incluindo variáveis usadas em declarações de Recode é 200.
2. Número máximo de variáveis preditoras (controle) por análise é 50.
3. Não é possível usar o número máximo de preditores, cada um com o número máximo de categorias, em uma análise. Se um problema excede a memória disponível, uma mensagem de erro é impressa, e o programa salta até a próxima análise.
4. Número máximo de análises por execução é 50.
5. Variáveis preditoras para análise de classificação múltipla devem ser categorizadas, preferencialmente com 6 ou menos categorias. As categorias devem possuir códigos inteiros no intervalo 0-31. Casos com qualquer outro valor será eliminado da análise.
6. Variável preditora para análise de variância univariada dev ser codificada no intervalo 0-2999. Casos com quaisquer outros valores são descartados da análise.
7. Se uma variável preditora possui casas decimais, somente a parte inteira é utilizada.
8. Se a variável de ID é alfabética com largura > 4, somente os quatro primeiros caracteres são usados.

29.9 Exemplos

Exemplo 1. Análise de classificação múltipla usando quatro variáveis controle (preditoras): V7, V9, V12, V13, e a variável dependente V100; análises separadas serão executadas no dataset inteiro e em dois subconjuntos de casos.

```
$RUN MCA
$FILES
PRINT  = MCA1.LST
DICTIN = LAB.DIC                        arquivo Dictionário de entrada
DATAIN = LAB.DAT                        arquivo Dados de entrada
$SETUP
ALL RESPONDENTS TOGETHER
*                               (valores default obtidos para todos parâmetros)
DEPV=V100  CONV=(V7,V9,V12-V13)
$RUN MCA
$SETUP
INCLUDE V4=21,31-39
ONLY SCIENTISTS
*                               (valores default obtidos para todos parâmetros)
DEPV=V100  CONV=(V7,V9,V12-V13)
$RUN MCA
$SETUP
INCLUDE V4=41-49
ONLY TECHNICIANS
*                               (valores default obtidos para todos parâmetros)
DEPV=V100  CONV=(V7,V9,V12-V13)
```

Exemplo 2. Análise de classificação múltipla com variável dependente V201 e três variáveis preditoras V101, V102, V107; dados devem ser ponderados pela variável V6; produzindo dataset de resíduos onde casos são identificados pela variável V2; casos com valores extremos (outliers de mais do que 4 desvios-padrões em relação a grand média) na variável dependente devem ser excluídos da análise. Resíduos para os primeiros 20 casos são listados subseqüentemente ao uso do program LIST.

```
$RUN MCA
$FILES
PRINT   = MCA2.LST
DICTIN  = LAB.DIC                       arquivo Dictionário de entrada
DATAIN  = LAB.DAT                       arquivo Dados de entrada
DICTOUT = LABRES.DIC                    arquivo Dictionário para resíduos
DATAOUT = LABRES.DAT                    arquivo Dados para resíduos
$SETUP
MULTIPLE CLASSIFICATION ANALYSIS  -  RESIDUALS WRITTEN INTO A FILE
*                               (valores default obtidos para todos parâmetros)
DEPV=V201  OUTL=EXCL  OUTD=4  IDVA=V2  WRITE=RESI  -
   CONV=(V101,V102,V107)  WEIGHT=V6
$RUN LIST
$SETUP
LISTING START OF RESIDUAL FILE
MAXCASES=20  INFILE=OUT
```

Exemplo 3. Para uma variável dependente V52, interações entre três variáveis (V7, V9, V12) serão checadas. V7 é codificada 1,2,9, V9 é codificada 1,3,5,9 e V12 é codificada 0,1,9 onde 9's são valores perdidos. Uma única variável de combinação é construída usando-se Recode. Isso envolve recodificar cada variável em um conjunto de códigos contíguos começando de zero e, então, usando a função COMBINE para produzir um único código para cada combinação possível de códigos das três variáveis separadas. MCA é executado usando 3 variáveis separadas como preditores e uma análise de variância univariada é executada usando a combinação de variáveis como controle. Casos com dados perdidos nos preditores serão excluídos. Casos com valores maiores do que 90000 na variável dependente serão também excluídos.

```
$RUN MCA
$FILES
DICTIN = CON.DIC                          arquivo Dictionário de entrada
DATAIN = CON.DAT                          arquivo Dados de entrada
$SETUP
EXCLUDE V7=9 OR V9=9 OR V12=9
CHECKING INTERACTIONS
BADD=SKIP
DEPV=(V52,90000)  CONVARS=(V7,V9,V12)
DEPV=(V52,90000)  CONVARS=R1
$RECODE
   R7=V7-1
   R9=BRAC(V9,1=0,3=1,5=2)
   R1=COMBINE R7(2),R9(3),V12(2)
```

Capítulo 30

Análise de Variância Multivariada (MANOVA)

30.1 Descrição Geral

MANOVA executa análise de variância univariada e multivariada, bem como análise de covariância, utilizando um modelo linear geral. Até oito fatores (variáveis independentes) podem ser usados. Se mais de uma variável dependente for especificada, tanto a análise univariada quanto a multivariada é executada. O programa aceita números iguais ou desiguais de casos nas células.

MANOVA é o único programa do IDAMS para análise de variância multivariada. ONEWAY é recomendado para análise de variância univariada. MCA lida com problemas univariados com multifatores. Ele não possui limitações no que diz respeito às células vazias, aceita mais do que 8 preditores, e permite mais do que 80 células. No entanto, o modelo analítico básico do MCA é diferente daquele do MANOVA. Uma diferença importante é que o MCA é insensível aos efeitos de interação.

Modelo de regressão hierárquica. MANOVA use uma abordagem via regressão para a análise de variância. Mais especificamente, o programa emprega um modelo hierárquico. Há uma importante conseqüência para o usuário: se uma execução do MANOVA envolve mais de uma variável fator, e se há números desproporcionais de casos nas células formadas pela cross-classificação dos fatores, então se deve considerar em que ordem as variáveis fator foram especificadas. Desproporcionalidade de números de subclasses confunde os efeitos principais e o pesquisador deve então escolher a ordem em que os efeitos confundidos devem ser eliminados. Ao usar MANOVA, essa escolha é alcançada pela ordem em que as variáveis fator são especificadas. Ao utilizar um ordenamento padrão, variáveis, inicialmente na especificação, possuem os efeitos de variáveis que são removidas posteriormente, e.g. o primeiro efeito listado será testado com todos os outros efeitos eliminados. A regra geral é que cada teste elimina efeitos listados antes dele, nas especificações do nome do teste, e ignora efeitos listados depois disso. Para uma análise padrão univariada, o termo de interação não é afetado pela ordem das variáveis fator; de maneira geral, para uma análise n-variadas, o enésimo termo de interação, e somente ele, não é afetado. O problema existe tanto para análise univariada quanto para multivariada.

Opção de contraste. Duas opções estão disponíveis para estabelecer os contrastes (ver o parâmetro de fator CONTRAST). Contrastes nominais são gerados por default; eles são os desvios de costume das médias das linhas e colunas da grande média e a generalização dessas para os contrastes de interação. O programa pode também gerar contrastes de Helmert.

Aumento da soma dos quadrados intra-células. É possível aumentar a soma dos quadrados intra-células (termo do erro) usando estimativas ortogonais (ver o parâmetro AUGMENT). Isso permite que o programa seja usado para quadrados Latinos, bem como para ajuntamento de termos de interação com o erro.

Reordenamento e/ou ajuntamento de estimativas ortogonais. Um ordenamento convencional das estimativas dos efeitos ortogonais (e.g. média, C, B, A, BxC, AxC, AxB, AxBxC para designs de três fatores) está construído no programa para utilização padrão. Contudo, estimativas ortogonais podem ser rearranjadas em diferentes ordens (ver o parâmetro REORDER). Além disso, é possível ajuntar várias

estimativas ortogonais, como os vários termos de interação, para teste simultâneo ou para partição do agrupamento de estimativas ortogonais de um dado efeito em agrupamentos menores para teste em separado (ver o parâmetro de nome de teste DEGFR).

30.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar casos para a execução. Variáveis dependentes são selecionadas pelo parâmetro DEPVARS e covariates pelo parâmetro COVARS. Variáveis fator são especificadas em declarações de fator especiais.

Transformando dados. Declarações de Recode podem ser usadas. Note que somente valores inteiros (positive ou negativo) são aceitos para variáveis usadas como fatores.

Ponderando dados. Use de variáveis de ponderação não é aplicável.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar que valores de dados perdidos, se houver algum, devem ser usados para checar a existência de dados perdidos. Casos com códigos em qualquer uma das variáveis de entrada (dependente, covariate ou variáveis de fator) são excluídas. Isso pode acarretar muitos valores excluídos e se constitui em um problema potencial que deve ser considerado ao se planejar a análise.

30.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variável, e registros-C, se houver, somente para variáveis usadas na execução.

Médias de célula e N's. Para cada célula, N é impresso e a média para cada variável dependente e covariate. As médias não ajustadas para nenhum covariate. Células são marcadas consecutivamente, começando com "1 1" (para designs de 2 fatores) a despeito dos códigos verdadeiros das variáveis de fator. Na indexação das células, os índices do último fator são os menores (se movem mais rápido).

Bases de design. Trata-se de uma matriz de design gerada pelo programa. As equações de efeitos estão em colunas, começando com o efeito da média na coluna 1. Se REORDER foi especificado, a matriz é impressa após o reordenamento.

Intercorrelações entre os coeficientes das equações normais.

Matriz de correlação de erro. Em uma análise de variância multivariada, o termo de erro é uma matriz de variância-covariância. Tal consiste naquele termo de erro (antes do ajustamento dos covariates, se houver algum) reduzido a uma matriz de correlação.

Componentes principais da matriz de correlação do erro. Os componentes estão em colunas. São os componentes do termo de erro (antes do ajustamento para os covariates, se houver algum) da análise.

Matriz de dispersão de erro e erros-padrão da estimação. Esse é o termo de erro, uma matriz de variância-covariância, para a análise. A matriz é ajustada para covariates, se houver algum. Cada elemento da diagonal da matriz é exatamente o que apareceria em uma tabela de análise de variância convencional como o erro quadrado médio "intra" para a variável. Graus de liberdade são ajustados para o processo de expansão se isso for requerido. Erros-padrão de estimação correspondem às raízes quadradas dos elementos da diagonal da matriz.

Para análise com covariate(s)

Matriz de dispersão de erro ajustada às correlações. Esse é o termo do erro, uma matriz de variância-covariância, depois dos ajustamentos para os covariates, reduzido a uma matriz de correlação.

Resumo de análise de regressão.

Componentes principais da matriz de correlação de erro depois de ajustamentos nos covariates. Os componentes estão em colunas. São os componentes do termo de erro da análise depois dos ajustamentos nos covariates.

Para análise univariada

Uma tabela anova. Graus de liberdade, soma de quadrados, quadrados médios e F-ratios.

Para análise multivariada

Os seguintes itens são impressos para cada efeito. Ajustamentos são feitos para covariates, se houver. A ordem dos efeitos é exatamente o oposto da ordem das especificações do nome do teste.

F-ratio para o critério de verossimilhança. A aproximação de Rao é utilizada. Isso é um teste de significância multivariada do efeito geral para todas as variáveis dependentes simultaneamente.

Variâncias canônicas dos componentes principais da hipótese. São as raízes, ou valores proprios, da matriz de hipótese.

Coeficientes dos componentes principais da hipótese. São as correlações entre as variáveis e os componentes da matriz de hipótese. O número de componentes não nulos para qualquer efeito será o mínimo entre os graus de liberdade e o número de variáveis dependentes.

Escores dos componentes de contraste para efeitos estimados. São os escores da hipótese para o contraste usado no design. Eles são análogos às médias das colunas em uma análise de variância univariada e podem ser utilizados da mesma maneira para localizar variáveis e contrastes que propiciem desvios não usuais da hipótese nula.

Teste cumulativo de Bartlett nas raízes. Esse é um teste aproximado para as raízes remanescentes, após eliminar a primeira, segunda, terceira, etc.

F-ratios para testes univariados. Esses são exatamente os F-ratios que seriam obtidos em uma análise univariada convencional.

30.4 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis de análise devem ser numéricas. A(s) variável(eis) e covariate(s) deve(m) ser medido(s) em uma escala de intervalo ou deve(m) ser dicotômica(s). As variáveis fator podem ser nominal, ordinal ou em intervalos, mas devem possuir valores inteiros; elas são utilizadas para designar a própria célula para o caso.

30.5 Estrutura de Setup

```
$RUN MANOVA

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de fator
        (repetidas como requisitadas; pelo menos um deve ser oferecido)
     5. Especificações de nome de teste
        (repetidas como requisitadas; pelo menos um deve ser oferecido)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default  IDAMS.LST)
```

30.6 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-5 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V2=1-4 AND V15=2
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  ANÁLISE DA IDADE E SALARIO COM SEXO E PROFISSAO COMO FATORES
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  DEPVARS=(V5,V8)  COVA=(V101,V102)
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada. Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
O número máximo de casos (depois da filtragem) a ser usado do arquivo de entrada.
Default: Todos os casos serão utilizados.

MDVALUES=**BOTH**/MD1/MD2/NONE
Quais valores de dados perdidos devem ser utilizados para variáveis acessadas nesse conjunto de tabelas. Ver o capítulo "O Arquivo Setup do IDAMS".

DEPVARS=(lista de variáveis)
Uma lista de variáveis a serem utilizadas como variáveis dependentes
Não há default.

COVARS=(lista de variáveis)
Uma lista de variáveis a serem utilizadas como covariates.

AUGMENT=(m,n)
Para formar o termo de erro, a soma dos quadrados intra será expandida pelas colunas m, m+1, m+2, ..., n da matriz de estimativas ortogonais.
Default: A soma dos quadrados intra será utilizada como o termo de erro.

REORDER=(lista de valores)
Reordena as estimativas ortogonais de acordo com a lista (ver o parágrafo "Reordenando e/ou ajuntando estimativas ortogonais", acima). Note que se o reordenamento das estimativas for requisitado, a ordem das especificações dos nomes dos testes deve corresponder à nova ordem.
Exemplo: O ordenamento convencional para um design de três fatores pode ser mudado para a ordem: média, A, B, C, AxB, AxC, BxC, AxBxC utilizando REORDER=(1,4,3,2,7,6,5,8).

PRINT=CDICT/DICT
CDIC Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT Imprime o dicionário de entrada sem registros-C.

4. **Especificações de fator** (pelo menos um deve ser oferecido). Até 8 especificações de fator devem ser oferecidas. As regras de codificação são as mesmas dos parâmetros. Cada especificação de fator deve iniciar em uma nova linha.

```
Exemplo:  FACTOR=(V3,1,2)
```

FACTOR=(número da variável, lista de valores de código)
Variável a ser utilizada como fator, seguida pelos valores do código que devem ser utilizados para designar a célula apropriada para o caso.

CONTRAST=**NOMINAL**/HELMERT
Especifica o tipo de contraste a ser utilizado na computação.
NOMI Contrastes nominais. Médias dos efeitos desviadas da grade média, i.e. M(1)-GM, M(2)-GM, etc.
HELM Contrastes de Helmer. Média do efeito 1 desviada da soma das medias 1 até r, onde r níveis estejam envolvidos.

5. **Especificações de nome de teste** (pelo menos um deve ser oferecido). Essas especificações identificam os testes que devem ser executados. Eles devem estar em uma ordem correta. Ordinariamente, haverá uma especificação para a grande média, seguida por uma especificação de nome para cada efeito principal, e finalmente, uma especificação de nome para cada interação possível. Se os parâmetros de design são reordenados ou os graus de liberdade são reagrupados (ver os parâmetros REORDER e DEGRF), as declarações de nome dos testes devem ser feitas de forma a se conformarem às modificações. As regras de codificação são as mesmas dos parâmetros. Cada especificação de nome de teste deve iniciar em uma nova linha.

```
Exemplo:  TESTNAME='grande média'
```

TESTNAME='nome de teste'
Nome com até 12 caracteres para cada teste deve ser fornecido. Primes são obrigatórios se o nome contiver caracteres não-alfanuméricos.

DEGFR=n
O agrupamento natural dos graus de liberdade (equações de parâmetro da hipótese) ocorre quando a ordem convencional dos testes estatísticos é utilizada. DEGFR é utilizado somente para modificar o agrupamento, e.g. quando você quer ajuntar vários termos de interação e testá-los simultaneamente, ou particionar os graus de liberdade de algum efeito em duas ou mais partes. Ao utilizar o parâmetro GEGFR, tenha certeza de usá-lo em todas as declarações de nome de teste, incluindo um grau de liberdade para a grande média.
Default: Se usa o agrupamiento natural dos graus de liberdade.

30.7 Restrições

1. O número máximo de variáveis dependentes é 19.
2. O número máximo de covariates é 20.
3. O número máximo de especificações de fator é 8.
4. O número máximo de valores de código em uma especificação de fator é 10.
5. O número máximo de células é 80.
6. Células com freqüência zero, com apenas um caso, ou com múltiplos casos idênticos, às vezes causam problemas; a execução termina prematuramente, ou ela pode terminar, mas produzir F-ratios inválidos e outras estatísticas.

30.8 Exemplos

Exemplo 1. Análise de variância univariada (V10 é a variável dependente) com dois fatores representados por A com códigos 1,2,3 e B com códigos 21 e 31; contrastes nominais serão utilizados nos cálculos, e testes serão executados na ordem convencional.

```
$RUN MANOVA
$FILES
PRINT   = MANOVA1.LST
DICTIN  = CM-NEW.DIC                    arquivo Dicionário de entrada
DATAIN  = CM-NEW.DAT                    arquivo Dados de entrada
$SETUP
ANALISE DE VARIANCIA UNIVARIADA
DEPVARS=v10
FACTOR=(V3,1,2,3)
FACTOR=(V8,21,31)
TESTNAME='grande média'
TESTNAME=B
TESTNAME=A
TESTNAME=AB
```

Exemplo 2. Análise de variância multivariada (V11 ? V14 são variáveis dependentes) com dois fatores ("sexo" codificado 1,2 e "idade" codificado 1,2,3); contrastes nominais serão utilizados nos cálculos, e testes serão executados na ordem convencional.

```
$RUN MANOVA
$FILES
     como no Exemplo 1
$SETUP
ANALISE DE VARIANCIA MULTIVARIADA
DEPVARS=(v11-v14)
FACTOR=(V2,1,2)
FACTOR=(V5,1,2,3)
TESTNAME='grande média'
TESTNAME=idade
TESTNAME=sexo
TESTNAME='sexo & idade'
```

Exemplo 3. Análise de variância multivariada (V11-V14 são variáveis dependentes) com três fatores (A codificado 1,2, B codificado 1,2,3, C codificado 1,2,3,4); contrastes nominais serão utilizados nos cálculos, e testes serão executados em uma ordem modificada (média, A, B, AxB, C, AxC, BxC, AxBxC).

```
$RUN MANOVA
$FILES
     como no Exemplo 1
$SETUP
ANALISE MULTIVARIADA DE VARIANCIA - TESTES EM UMA ORDEM MODIFICADA
DEPVARS=(v11-v14) REORDER=(1,4,3,7,2,6,5,8)
FACTOR=(V2,1,2)
FACTOR=(V5,1,2,3)
FACTOR=(V8,1,2,3,4)
TESTNAME=média
TESTNAME=A
TESTNAME=B
TESTNAME=AxB
TESTNAME=C
TESTNAME=AxC
TESTNAME=BxC
TESTNAME=AxBxC
```

Capítulo 31

Análise de Variância Univariada (ONEWAY)

31.1 Descrição Geral

ONEWAY é um program de análise de variância univariada. Um número ilimitado de tabelas, usando vários pares de variáveis independentes e dependentes, podem ser produzidas em uma só execução. Cada análise pode ser executada em todos os casos ou em subconjuntos de casos do arquivo de dados; a seleção de caso para uma análise é independente da seleção para outras análises. O termo "variável de controle" usado em ONEWAY é equivalente a "variável independente", "preditor" ou, em terminologia de análise de variância, "variável de tratamento".

Uma alternativa a ONEWAY é o programa MCA quando apenas um preditor é especificado. Ele permite um código máximo de 2999 para uma variável de controle, enquanto que ONEWAY é limitado a um código máximo de 99.

31.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto dos casos dos dados de entrada. Esse filtro afeta todas as análises em uma execução. Além disso, até dois filtros locais estão disponíveis para selecionar independentemente um subconjunto dos casos de dados para cada análise. Se dois filtros locais são utilizados, um caso deve satisfazer ambos para ser incluído na análise. Variáveis são selecionadas para cada análise pelo parâmetro de tabela DEPVARS e CONVARS. Uma tabela separada é produzida para cada variável da lista DEPVARS com cada variável da lista CONVARS.

Transformando dados. Declarações de Recode podem ser usadas.

Ponderando dados. Uma variável pode ser usada para ponderar os dados de entrada; essa variável de ponderação pode ter valores inteiros ou decimais. Quando o valor da variável de ponderação é zero, negativo, perdido ou não-numérico, então o caso é sempre evitado; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos. O parâmetro de tabela MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser usados para checar a existência de dados perdidos. Casos com dados perdidos na variável dependente são sempre excluídos. Casos com dados perdidos na variável de controle podem, opcionalmente, ser excluídos (ver o parâmetro de tabela MDHANDLING).

31.3 Resultados

Especificações de tabela. Uma lista de especificações de tabela oferecendo uma lista de conteúdos para os resultados.

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, somente para variáveis usadas na execução.

Estatísticas descritivas dentro de categorias da variável de controle. Estatísticas intermediárias são impressas em formato de tabela para cada valor de código da variável de controle, mostrando:

o número de casos válidos (N) e a soma dos pesos (arredondados para o inteiro mais próximo),
soma dos pesos como percentual da soma total,
média, desvio-padrão, coeficiente de variação, soma e soma dos quadrados da variável dependente,
soma da variável dependente como percentual da soma total.

Uma linha de totais é impressa para a tabela dando somas ao longo de todas as categorias da variável de controle (exceto categorias com zero graus de liberdade, que são excluídas dos totais).

Estatísticas de análise de variância. Categorias da variável de controle que possuem zero graus de liberdade não são incluídas na computação dessas estatísticas. As seguintes estatísticas são incluídas para cada tabela:

soma total de quadrados da variável dependente,
eta e eta quadrado (não-ajustado e ajustado),
a soma de quadrados entre grupos (soma de quadrados entre médias) e soma de quadrados dentro dos grupos,
o F-ratio (somente impresso se os dados são não-ponderados).

31.4 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis de análise devem ser numéricas; elas podem assumir valores inteiros ou decimais.

Uma variável dependente deve ser medida em uma escala de intervalo ou ser dicotômica. Uma variável de controle pode ser nominal, ordinal ou intervalo, mas deve possuir valores no intervalo 0-99. Se, para qualquer caso, a variável de controle para uma análise possuir um valor excedendo esse intervalo, o caso é eliminado daquela análise; nenhuma mensagem é dada. Se o valor da variável de controle possui casas decimais, somente a parte inteira é usada (e.g. 1.1 e 1.6 são ambos colocados no lugar 1); nenhuma mensagem é dada.

31.5 Estrutura de Setup

```
$RUN ONEWAY

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de tabela (repetidas como requisitadas)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados

Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default  IDAMS.LST)
```

31.6 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-4 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  EXCLUDE V3=9
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  DATA ON TRAINING EFFECTS FOR FOOTBALL PLAYERS
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  *
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada. Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

   MAXCASES=n
   O número máximo de casos (depois da filtragem) a ser usado do arquivo de entrada. Default: Todos os casos serão utilizados.

PRINT=CDICT/DICT
- CDIC Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
- DICT Imprime o dicionário de entrada sem registros-C.

4. **Especificações de tabela.** As regras de codificação são as mesmas das de parâmetros. Cada especificação de tabela deve começar em uma nova linha.

```
Exemplos:  CONV=V6 DEPV=V26 WEIG=V3 F1=(V14,2,7) F2=(V13,1,1)
           CONV=V5 DEPV=(V27-V29,V80)
```

DEPVARS=(lista de variáveis)
Uma lista de variáveis a serem utilizadas como variáveis dependentes

CONVARS=(lista de variáveis)
Uma lista de variáveis a serem utilizadas como variáveis de controle.

WEIGHT=número de variável
O número da variável de ponderação se os dados devem ser ponderados.

MDVALUES=**BOTH**/MD1/MD2/NONE
Quais valores de dados perdidos devem ser utilizados para variáveis acessadas nesse conjunto de tabelas. Ver o capítulo "O Arquivo Setup do IDAMS".

MDHANDLING=**DELETE**/KEEP
- DELE Deleta casos com dados perdidos na variável de controle.
- KEEP Inclui casos com dados perdidos na variável de controle.

Nota: Casos com dados perdidos na variável dependente são sempre deletados.

F1=(número de variável, código válido mínimo, código válido máximo)
F1 se refere a primeira variável de filtro que é usada para criar um subconjunto dos dados. O número da variável deve ser o número da variável de filtro; casos cujos valores para essa variável caem no intervalo mínimo-máximo serão incluídas na tabela. O valor mínimo pode ser um inteiro negativo. O número máximo deve ser menor do que 99.999. Casas decimais devem ser colocadas onde apropriadas.

F2=(número de variável, código válido mínimo, código válido máximo)
F2 se refere à segunda variável de filtro. Se esse segundo filtro é especificado, um caso deve satisfazer os requerimentos de ambos para entrar na tabela.

31.7 Restrições

1. O número máximo de variáveis de controle é 99. O número máximo de variáveis dependentes é 99. O número total de variáveis que podem ser acessadas é 204, incluindo variáveis usadas em declarações de Recode.

2. ONEWAY usa os valores da variável de controle no intervalo de 0 a 99. Se, para quallquer caso, a variável de controle para uma certa análise possuir um valor excedendo esse intervalo, o caso é eliminado daquela tabela.

3. A soma máxima de pesos é aproximadamente 2.000.000.000.

4. O F-ratio é impresso apenas para dados não-ponderados.

31.8 Exemplos

Exemplo 1. Três análises de variância univariada usando V201 como controle e V204 como variável dependente: primeiro para o conjunto inteiro, segundo para um subconjunto de casos tendo valores 1-3 para a variável V5, e o terceiro subconjunto de casos possuindo valores 4-7 para a variável V5.

```
$RUN ONEWAY
$FILES
PRINT  = ONEW1.LST
DICTIN = STUDY.DIC                      arquivo Dicionário de entrada
DATAIN = STUDY.DAT                      arquivo Dados de entrada
$SETUP
ONE-WAY ANALYSES OF VARIANCE DESCRIBED SEPARATELY
*                       (valores default obtidos para todos parâmetros)
CONV=V201  DEPV=V204
CONV=V201  DEPV=V204  F1=(V5,1,3)
CONV=V201  DEPV=V204  F1=(V5,4,7)
```

Exemplo 2. Geração de uma análise de variância univariada para todas as combinações de variáveis de controle V101, V102, V105 e V110, e variáveis dependentes V17 até V21; dados são ponderados pela variável V3.

```
$RUN ONEWAY
$FILES
    como no Exemplo 1
$SETUP
MASS-GENERATION OF ONE-WAY ANALYSES OF VARIANCE
*                       (valores default obtidos para todos parâmetros)
CONV=(V101,V102,V105,V110)  DEPV=(V17-V21)  WEIGHT=V3
```

Capítulo 32

Scoring Baseado em Ordenação Parcial de Casos (POSCOR)

32.1 Descrição Geral

POSCOR calcula (escala ordinal) escores usando um procedimento baseado na posição hierárquica dos elementos em um conjunto parcialmente ordenado de acordo com um número de propriedades (ou características, etc.). Os escores, calculados separadamente para cada elemento do conjunto, são processados em uma arquivo Dados descrito por um dicionário do IDAMS. Esse arquivo pode então ser usado como entrada em outros programas de análise.

Usando o parâmetro ORDER, tipos diferentes de escores podem ser obtidos: (1) quatro tipos de escores onde os cálculos são baseados na proporção de casos dominados pelo caso examinado; (2) quatro outros escores onde cálculos são baseados na proporção de casos que dominam o caso examinado. O intervalo dos escores é determinado pelo parâmetro SCALE. Valores de escores significativos podem ser esperados somente quando o número de casos envolvidos for muito maior que o número de variáveis (ou componentes do escore) especificadas.

Em aplicações com variáveis de importância não uniforme, uma lista de prioridade pode ser definida usando o parâmetro de análise LEVEL no ordenamento parcial. Se as variáveis de prioridade mais alta determinam sem ambigüidade a relação de dois casos, as variáveis de prioridade mais baixa não são consideradas.

No caso especial quando apenas uma variável é usada em uma análise, os valores transformados correspondem às suas probabilidades (ver as opções ORDER=ASEA/DEEA/ASCA/DESA).

Em uma análise, uma série de subconjuntos mutuamente exclusivos pode ser examinada usando a facilidade do subconjunto. Nesse evento, a(s) variável(eis) de escore são computadas dentro de cada subconjunto de casos.

32.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar casos para a execução. Uma opção de subsetting de casos está disponível também para cada análise. Variáveis a serem transferidas para o arquivo de saída são selecionadas nas especificações de análise.

Transformando dados. Declarações de Recode podem ser usadas. Note que somente a parte inteira de variáveis recodificadas é usada pelo programa, i.e. variáveis recodificadas são arredondadas para o inteiro mais próximo.

Ponderando dados. Uso de variáveis de ponderação não é aplicável.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser usados para checar a existência de dados perdidos. O parâmetro MDHANDLING indica se variáveis ou casos com dados perdidos devem ser excluídos da análise.

32.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, somente para variáveis usadas na execução.

Dicionário de saída. (Opcional: ver o parâmetro PRINT).

32.4 Dataset de Saída

O arquivo de saída contém os escores computados juntamente com as variáveis transferidas e, opcionalmente, variáveis de análise, para cada caso usado na análise (i.e. todos os casos passando pelo filtro e não excluídos pelo uso da opção de manuseio de dados perdidos). Um dicionário do IDAMS associado é também produzido.

Variáveis de saída são numeradas seqüencialmente começando de 1 e possuem as seguintes características:

- *Variáveis de análise e de subconjunto* (opcional: somente se AUTR=YES). Variáveis-V possuem as mesmas características dos seus equivalentes de entrada. Variáveis de Recode são produzidas com WIDTH=7 e DEC=0.
- *Variáveis de identificação de caso (ID) e transferidas.* Variáveis-V possuem as mesmas características dos seus equivalentes de entrada. Variáveis de Recode são produzidas com WIDTH=7 e DEC=0.
- *Variáveis de escore computadas.* Para ORDER=ASEA/DEEA/ASCA/DESA, uma variável para cada análise com:

Nome	especificado por ANAME	(default: em branco)
Largura de campo	especificado por FSIZE	(default: 5)
No. de decimais	0	
MD1	especificado por OMD1	(default: 99999)
MD2	especificado por OMD2	(default: 99999)

  Para ORDER=ASER/DESR/ASCR/DEER, duas variáveis para cada análise com nomes especificados pelos parâmetros ANAME e DNAME, respectivamente, e outras características como evidenciado acima.

Nota. Se uma análise é repetida para vários subconjuntos mutuamente exclusivos de casos, a variável de escore é computada para os casos em cada subconjunto de cada vez. Se um caso não pertence a nenhum dos subconjuntos definidos para a análise, então o(s) valor(res) da(s) variável(veis) de escore será(ão) igualado(s) ao código MD1.

32.5 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Para variáveis de análise, apenas valores inteiros são usados. Valores decimais, se houver, são arredondados para o inteiro mais próximo. A variável de ID de caso e variáveis a serem transferidas podem ser alfabéticas.

32.6 Estrutura de Setup

```
$RUN POSCOR

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de subconjunto (opcional)
     5. POSCOR
     6. Especificações de análise (repetidas como requisitadas)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída
DATAyyyy   dados de saída
PRINT      resultados (default  IDAMS.LST)
```

32.7 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais detalhadas das declarações de controle de programa, itens 1-3 e 6 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V2=1-4 AND V15=2
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear resultados.

   ```
   Exemplo:  SCALING THE RU INPUT VARIABLES
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo: MDHAND=CASES  TRAN=V5 IDVAR=R6
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2
Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
O número máximo de casos (depois da filtragem) a serem usados do arquivo de entrada.
Default: Todos os casos serão utilizados.

MDVALUES=**BOTH**/MD1/MD2/NONE
Quais valores de dados perdidos devem ser usados para as variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".

MDHANDLING=**VARS**/CASES
Tratamento de dados perdidos.

VARS	Uma variável contendo um valor de dados perdidos é excluída da comparação.
CASE	Uma variável contendo um valor de dados perdidos é excluída da análise.

OUTFILE=**OUT**/yyyy
Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
Default ddnames: DICTOUT, DATAOUT.

IDVAR=número de variável
Variável a ser transferida para o dataset de saída para identificar os casos.
Não há default.

TRANSVARS=(lista de variáveis)
Variáveis adicionais (até 99) a serem transferidas para o dataset de saída. Essa lista não deve incluir variáveis de análise ou variáveis usadas em especificações de subconjunto. Essas são transferidos automaticamente usando o parâmetro AUTR.

AUTR=**YES**/NO

YES	Variáveis de análise e variáveis usadas em especificações de subconjunto serão automaticamente transferidas para o dataset de saída.
NO	Nenhuma transferência de variáveis de análise ou de subconjunto.

FSIZE=**5**/n
Largura do campo das variáveis (escores) computadas.

SCALE=**100**/n
O valor (fator de escala) especificando o intervalo (0 - n) dos escores computados.

OMD1=**99999**/n
Valor do primeiro código de dados perdidos para as variáveis computadas (escores).

OMD2=**99999**/n
Valor do segundo código de dados perdidos para as variáveis computadas (escores).

PRINT=(CDICT/DICT, **OUTDICT**/OUTCDICT/NOOUTDICT)

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
OUTD	Imprime o dicionário de saída sem registros-C.
OUTC	Imprime o dicionário de saída com registros-C, se houver.
NOOU	Não imprime o dicionário de saída.

4. **Especificações de subconjunto** (opcional). Elas especificam subconjuntos de casos mutuamente exclusivos para uma análise particular.

```
Exemplo:  AGE      INCLUDE V5=15-20,21-45,46-64
```

Regras de codificação

Protótipo: nome declaração

nome
: Nome do subconjunto. 1-8 caracteres alfanuméricos iniciando-se com uma letra. Esse nome deve corresponder exatamente ao nome usado em especificações de análise subseqüentes. Espaços em branco intercalados não serão permitidos. É recomendável que todos os nomes sejam justificados a esquerda.

declaração
: Definição de subconjunto.
  - Comece com a palavra INCLUDE.
  - Especifique o número de variável (variável-V ou R) no qual os subconjuntos devem se basear (variáveis alfabéticas não são permitidas).
  - Especifique valores e/ou intervalos de valores separados por vírgulas. Cada valor ou intervalo define um subconjunto. Vírgulas separam os subconjuntos. Intervalos negativos devem ser expressos em seqüência numérica, e.g -4 - -2 (para -4 a -2); -2 - 5 (para -2 a +5). Os subconjuntos devem ser mutuamente exclusivos (i.e. um mesmo valor não pode aparecer em dois intervalos). No exemplo acima, 3 subconjuntos baseados no valor de V5 são definidos para a especificação de subconjunto de AGE.
  - Entre com um traço no final de uma linha para continuar em outra linha.

5. **POSCOR.** A palavra POSCOR nessa linha sinaliza que as especificações de análise seguem. Deve ser incluída (para separar especificações de subconjuntos das especificações de análise) e deve aparecer somente uma vez.

6. **Especificações de análise.** As regras de codificação são as mesmas dos parâmetros. Cada especificação de análise deve se iniciar em uma nova linha.

```
Exemplo:  ORDER=ASER  ANAME=MSDCORE  DNAME=DOWNSCORE  -
              VARS=(V3-V6)  LEVELS=(1,1,2,2)
```

VARS=(lista de variáveis)
: As variáveis-V e/ou -R a serem usadas na análise.
Não há default.

ORDER=**ASEA**/DEEA/ASCA/DESA/ASER/DESR/ASCR/DEER
: Especifica o tipo de escore a ser computado.
O escore é baseado em:

ASEA	casos melhores ou iguais/dominando
DEEA	casos piores ou iguais/dominados
ASCA	casos estritamente melhores/ estritamente dominando
DESA	casos estritamente piores ou iguais/estritamente dominados
	relativamente ao número **total** de casos

ASER/DESR

ASER	casos melhores ou iguais/dominando
DESR	casos estritamente piores ou iguais/estritamente dominados
	relativamente ao número de casos **comparáveis**

ASCR/DEER

ASCR	casos estritamente melhores/ estritamente dominando
DEER	casos piores ou iguais/dominados
	relativamente ao número de casos **comparáveis**

Nota. Em ambos os casos anteriores os dois escores são computados, seja qual for o selecionado. A soma deles é igual ao valor especificado no parâmetro SCALE.

SUBSET=xxxxxxxx
: Especifica o nome da especificação de subconjunto a ser utilizada, se houver. Coloque o nome entre aspas simples se ele contiver caracteres não-alfanuméricos. Letras maiúsculas devem ser usadas para coincidir com o nome da especificação de subconjunto que é automaticamente convertida para maiúsculas.

LEVELS=**(1, 1,..., 1)** / (N1, N2, N3,...,Nk)
: "k" é o número de variáveis usadas na lista de variáveis de análise. Ni define a ordem de prioridade da i'ésima variável na lista de variáveis envolvidas no ordenamento parcial. Um valor maior implica uma prioridade menor. Os valores de prioridade devem ser especificados na mesma seqüência das variáveis correspondentes na lista de variáveis de análise. O default de 1's implica que todas as variáveis possuem a mesma prioridade.

ANAME='nome'
: Um nome de até 24 caracteres para o escore crescente. Aspas simples são mandatórias se o nome contiver caracteres não-alfanuméricos.
Default: Em branco.

DNAME='nome'
: Um nome de até 24 caracteres para o escore decrescente. Aspas simples são mandatórias se o nome contiver caracteres não-alfanuméricos.
Default: Em branco.

32.8 Restrições

1. Os valores das variáveis de análise devem estar entre -32.767 e +32.767.
2. Os componentes da lista de prioridade no parâmetro LEVEL devem ser inteiros entre 1 e 32.767.
3. Número máximo de análises é 10.
4. Número máximo de variáveis a serem transferidas é 99.
5. Uma variável só pode ser usada uma vez se ela for uma variável de ID, em uma lista de análise ou lista de transferência. Se é requisitado usar a mesma variável duas vezes, então usa-se a recodificação para obter-se uma cópia com um número de variável (resultado) diferente.
6. O número máximo de variáveis usadas para análise, em especificações de subconjunto e em uma lista de transferência é 100 (incluindo tanto variáveis-V quanto -R).
7. Número máximo de especificações de subconjunto é 10.
8. Se a variável de ID ou uma variável a ser transferida é alfabética com largura > 4, somente os quatro primeiros caracteres serão usados.
9. Apesar do número de casos processados não ser limitado, deve ser notado que o tempo de execução cresce como uma função quadrática do número de casos sendo analisados.

32.9 Exemplos

Exemplo 1. Computação de dois escores usando as variáveis V10, V12, V35 até V40; o primeiro escore será calculado com os dados completos, enquanto o segundo será calculado separadamente em três subconjuntos (para os valores 1,2 e 3 da variável V7); casos com dados perdidos devem ser excluídos da análise; ambos os escores são baseados nos casos estritamente dominados relativamente ao número de casos comparáveis; casos são identificados pelas variáveis V2 e V4 que são transferidas para o arquivo de saída. Note que Recode é usado para fazer uma cópia das variáveis, pois uma restrição no programa é a de que uma variável só pode ser utilizada uma vez em uma execução.

```
$RUN POSCOR
$FILES
PRINT   = POSCOR1.LST
DICTIN  = PREF.DIC                      arquivo Dicionário de entrada
DATAIN  = PREF.DAT                      arquivo Dados de entrada
DICTOUT = SCORES.DIC                    arquivo Dicionário de saída
DATAOUT = SCORES.DAT                    arquivo Dados de saída
$SETUP
COMPUTATION OF TWO SCORES
MDHAND=CASES  IDVAR=V2  TRANSVARS=V4
TYPE     INCLUDE V7=1,2,3
POSCOR
ORDER=DESR  ANAME='GLOBAL SCORE  INCR'  DNAME='GLOBAL SCORE  DECR' -
   VARS=(V10,V12,V35-V40)
ORDER=DESR  ANAME='ADJUSTED SCORE   INCR' -
   DNAME='ADJUSTED SCORE   DECR'   SUBS=TYPE -
   VARS=(R10,R12,R35-R40)
$RECODE
   R10=V10
   R12=V12
   R35=V35
   R36=V36
   R37=V37
   R38=V38
   R39=V39
   R40=V40
```

Exemplo 2. Cômputo de três escores baseados em casos dominantes relativos ao número total de casos; variáveis de análise não devem ser transferidas para o arquivo de saída; variáveis contendo valores de dados perdidos devem excluídas da comparação; variáveis de identificação de caso V1 e V5 são transferidas.

```
$RUN POSCOR
$FILES
     como no Exemplo 1
$SETUP
COMPUTATION OF THREE SCORES
AUTR=NO  IDVAR=V1  TRANSVARS=V5
POSCOR
ORDER=ASEA  ANAME='SCORE 1   INCR'  VARS=(V11,V17,V55-V60)
ORDER=ASEA  ANAME='SCORE 2   INCR'  VARS=(V108-V110,V114,V116,V118,V120)
ORDER=ASEA  ANAME='SCORE 3   INCR'  VARS=(V22,V33,V101-V105)
```

Capítulo 33

Correlação de Pearson (PEARSON)

33.1 Descrição Geral

PEARSON computa e imprime a matriz de coeficiente de correlação r de Pearson e covariâncias para todos os pares de variáveis em uma lista (opção matriz quadrada) ou para cada par de variáveis formado ao se obter uma variável de cada uma das duas listas (opção matriz retangular).

Tanto a deleção por pares ou por casos de dados perdidos pode ser especificada.

PEARSON pode ser também utilizado para produzir uma matriz de correlação que pode subseqüentemente ser entrada para os programas REGRESSN ou MDSCAL. Apesar de REGRESSN ser capaz de computar sua própria matriz de correlação, seu manuseio de dados perdidos é limitado à deleção por pares. Contrastando, uma matriz pode ser gerada por PEARSON usando-se um algoritmo de deleção por pares para dados perdidos.

33.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. As variáveis de cujas correlações são desejadas são especificadas nos parâmetros ROWVARS e COLVARS.

Transformando dados. Declarações de Recode podem ser usadas.

Ponderando dados. Uma variável pode ser usada para ponderar dados de entrada; essa variável de ponderação pode ter valores inteiros e decimais. Quando o valor da variável de ponderação para um caso for zero, negativo, perdido ou não-numérico, então o caso será sempre evitado; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houverem, devem ser usados para checar a existência de dados perdidos. As estatísticas univariadas para cada variável são computadas dos casos que possuem dados válidos (não perdidos) para a variável.

Dados perdidos: deleção por pares. Estatísticas emparelhadas e cada coeficiente de correlação podem ser computadas dos casos que possuem dados válidos para ambas as variáveis (MDHANDLING=PAIR). Portanto, um caso pode ser usado no cômputo de alguns pares de variáveis e não ser usado em outros pares. Esse método de manusear dados perdidos é referido como o algoritmo de deleção por pares. Nota: Se há dados perdidos, coeficientes de correlação individuais podem ser computados em diferentes subconjuntos dos dados. Se há uma grande quantidade de dados perdidos, isso pode levar à inconsistências internas na matriz de correlação que podem causar dificuldades em análises multivariadas subseqüentes.

Dados perdidos: deleção por casos. O programa pode também ser instruído (MDHANDLING=CASE) a computar estatísticas emparelhadas e correlações dos casos que possuem dados válidos em todas as variáveis na lista de variáveis. Portanto, um caso é usado no cômputo para todos os pares de variáveis ou não é usado

de jeito nenhum. Esse método de manusear dados é referido como o algoritmo de deleção por casos (também disponível no programa REGRESSN), e se aplica somente à opção de matriz quadrada.

33.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, apenas para variáveis usadas na execução.

Opção de matriz quadrada

Estatísticas emparelhadas. (Opcional: ver o parâmetro PRINT). Para cada par de variáveis na lista de variáveis o seguinte é impresso:
número de casos válidos (ou soma ponderada de casos),
média e desvio-padrão da variável X,
média e desvio-padrão da variável Y,
t-test para coeficiente de correlação,
coeficiente de correlação.

Estatísticas univariadas. Para cada variável na lista de variáveis, o seguinte é impresso:
número de casos válidos e soma dos pesos,
soma dos escores e soma dos escores ao quadrado,
média e desvio-padrão.

Coeficientes de regressão para escores brutos. (Opcional: ver o parâmetro PRINT). Para cada par de variáveis x e y, os coeficientes a e c e os termos constantes b e d nas equações de regressão x=ay+b e y=cx+d são impressos.

Matriz de correlação. (Opcional: ver o parâmetro PRINT). A parte triangular inferior esquerda da matriz.

Matriz de produtos cruzados. (Opcional: ver o parâmetro PRINT). A parte triangular inferior esquerda da matriz.

Matriz de covariância. (Opcional: ver o parâmetro PRINT). A parte triangular inferior esquerda da matriz com a diagonal.

Em cada uma das matrizes acima, um máximo de 11 colunas e 27 linhas são impressas por página.

Opção de matriz retangular

Tabela de freqüências de variáveis. Número de casos válidos para cada par de variáveis.

Tabela de valores médios para variáveis de coluna. Médias são calculadas e impressas para cada variável de coluna ao longo de todos os casos que sejam válidos para cada variável de linha.

Tabela de desvios-padrões para variáveis de coluna. Igual às médias.

Matriz de correlação. (Opcional: ver o parâmetro PRINT). Coeficientes de correlação para todos os pares de variáveis.

Matriz de covariância. (Opcional: ver o parâmetro PRINT). Covariâncias para todos os pares de variáveis.

Em cada uma das matrizes acima, um máximo de 8 colunas e 50 linhas são impressas por página.

Nota: Se um par de variáveis não possui casos válidos, 0.0 é impresso para a média, desvio-padrão, correlação e covariância.

33.4 Matrizes de Saída

Matriz de correlação

A matriz de correlação na forma de matriz quadrada do IDAMS é produzida quando o parâmetro WRITE=CORR é especificado. O formato usado para escrever as correlações é 8F9.6; o formato para ambas médias e desvios-

padrões, é 5E14.7. Colunas 73-80 são utilizadas para identificar os registros.

A matriz contém correlações, médias, e desvios-padrões. As médias e desvios-padrões estão desemparelhados. Os registros de dicionário que são produzidos por PEARSON contém números e nomes de variáveis do dicionário de entrada e/ou declarações de Recode. A ordem das variáveis é determinada pela ordem das variáveis na lista de variáveis.

PEARSON pode gerar correlações iguais a 99.99901, e médias e desvios-padrões iguais a 0.0 quando não for possível computar um valor inteligível. Razões típicas para isso são quando todos os dados são eliminados devido a dados perdidos ou uma das variáveis possui um valor constante. Note que MDSCAL não aceita esses "valores perdidos", apesar de REGRESSN aceitar.

Matriz de covariância

A matriz de covariância sem a diagonal na forma de uma matriz quadrada do IDAMS é produzida quando o parâmetro WRITE=COVA é especificado.

33.5 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis de análise devem ser numéricas; elas podem ter valores inteiros ou decimais.

33.6 Estrutura de Setup

```
$RUN PEARSON

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos :
FT02       matrizes de saída se parâmetro WRITE especificado
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default  IDAMS.LST)
```

33.7 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

```
Exemplo:  INCLUDE  V2=11-15,60  OR  V3=9
```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

```
Exemplo:  FIRST EXECUTION OF PEARSON - APRIL 27
```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

```
Exemplo:  WRITE=CORR, PRINT=(CORR,COVA)  ROWV=(V1,V3-V6,R47,V25)
```

INFILE=**IN**/xxxx
: Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2
: Tratamento de valores não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
: O número máximo de casos (depois da filtragem) a serem usados do arquivo de entrada.
Default: Todos os casos serão usados.

MATRIX=**SQUARE**/RECTANGULAR

SQUA	Computa os coeficientes de correlação de Pearson para todos os pares de variáveis da lista ROWV.
RECT	Computa os coeficientes de correlação de Pearson para todos os pares de variáveis formados ao obter-se uma variável de cada uma das listas ROWV e COLV.

ROWVARS=(lista de variáveis)
: Uma lista de variáveis-V e/ou -R a serem correlacionadas (MATRIX=SQUARE) ou a lista de variáveis de linha (MATRIX=RECTANGULAR).
Não há default.

COLVARS=(lista de variáveis)
: (MATRIX=RECTANGULAR apenas).
Uma lista de variáveis-V e/ou -R a serem usadas como variáveis de coluna. Oito colunas são impressas por página; se a lista de variáveis de linha ou a lista de variáveis de coluna contém menos do que oito variáveis, é preferível (para facilidade de leitura dos resultados) ter a lista curta como a lista de variável de coluna.

MDVALUES=**BOTH**/MD1/MD2/NONE
: Que valores de dados perdidos devem ser usados para as variáveis acessadas nessa execução. Ver o capítulo o "O Arquivo Setup do IDAMS".

MDHANDLING=**PAIR**/CASE
: Método de manuseio de dados perdidos.

PAIR	Deleção por pares.
CASE	Deleção por casos (não disponível com MATRIX=RECTANGULAR).

WEIGHT=número de variável
: O número da variável de ponderação se os dados forem ponderados.

WRITE=(CORR, COVA)
: (MATRIX=SQUARE apenas).

CORR	Produz a matriz de correlação com médias e desvios-padrões.
COVA	Produz a matriz de covariância com médias e desvios-padrões.

PRINT=(CDICT/DICT, **CORR**/NOCORR, COVA, PAIR, REGR, XPRODUCTS)

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
CORR	Imprime a matriz de correlação.
COVA	Imprime a matriz de covariância.
PAIR	Imprime as estatísticas emparelhadas (MATRIX=SQUARE apenas).
REGR	Imprime os coeficientes da regressão (MATRIX=SQUARE apenas).
XPRO	Imprime a matriz de produtos cruzados (MATRIX=SQUARE apenas).

33.8 Restrições

Quando MATRIX=SQUARE é especificado

1. O número máximo de variáveis permitidas em uma execução é 200. Esse limite inclui todas as variáveis de análise, e variáveis usadas em declarações de Recode.
2. Números de variáveis de Recode não devem exceder 999 se o parâmetro WRITE é especificado. (Elas são produzidas como números negativos na parte descritiva da matriz que tenha apenas 4 colunas reservadas ao número da variável e.g R862 se torna -862).

Quando MATRIX=RECTANGULAR é especificado

1. O número máximo de variáveis em uma lista de variáveis de linha ou coluna é 100.
2. O número total máximo de variáveis de linha, variáveis de coluna, variáveis usadas em declarações de Recode, e a variável de ponderação é 136.

33.9 Exemplos

Exemplo 1. Cálculo de uma matriz quadrada de coeficientes de correlação r de Pearson com deleção por pares de casos que possuam dados perdidos; a matriz será escrita em um arquivo e impressa.

```
$RUN PEARSON
$FILES
PRINT  = PEARS1.LST
FT02   = BIRDCOR.MAT                    arquivo Matriz de saída
DICTIN = BIRD.DIC                       arquivo Dicionário de entrada
DATAIN = BIRD.DAT                       arquivo Dados de entrada
$SETUP
MATRIX OF CORRELATION COEFFICIENTS
PRINT=(PAIR,REGR,CORR)  WRITE=CORR  ROWV=(V18-V21,V36,V55-V61)
```

Exemplo 2. Cálculo de coeficientes de correlação r de Pearson para as variáveis V10-V20 com as variáveis V5-V6.

```
$RUN PEARSON
$FILES
DICTIN = BIRD.DIC                       arquivo Dicionário de entrada
DATAIN = BIRD.DAT                       arquivo Dados de entrada
$SETUP
CORRELATION COEFFICIENTS
MATRIX=RECT  ROWV=(V10-V20)  COLV=(V5-V6)
```

Capítulo 34

Ordenamento de Alternativas (RANK)

34.1 Descrição Geral

RANK determina um ordenamento de alternativas razoável, usando dados de preferência como entrada e três procedimentos diferentes, um baseado em lógica clássica (o método ELECTRE) e dois outros baseados em lógica difusa. As duas abordagens diferem essencialmente na maneira em que as matrizes relacionais são construídas. Com o ordenamento difuso, os dados determinam completamente os resultados, enquanto com ordenamento clássico o usuário, amparando-se em conceitos de lógica clássica, tem a possibilidade de controlar os cálculos de todas as relações entre alternativas.

O método ELECTRE (lógica clássica) implementado em RANK, em um primeiro passo, usa os dados de preferência de entrada para calcular uma matriz final expressando a opinião coletiva geral sobre a "dominância" entre as alternativas, a estrutura da relação não necessariamente correspondendo a uma ordem linear ou parcial. A relação de "dominância" para cada par de alternativas é controlada pelas condições de "concordância" e "discordância" fixadas pelo usuário. Estruturas relacionais diferentes podem ser obtidas dos mesmos dados ao variarem-se os parâmetros de análise. Em um segundo passo, o procedimento busca uma seqüência de camadas não-dominadas (núcleos) de alternativas. O primeiro núcleo consiste de alternativas de maiores ranks ao longo de todo o conjunto considerado. Deve-se notar que, em certos casos, núcleos adicionais podem não existir devido a loops na relação. Isso pode ser verdadeiro mesmo no nível mais alto.

O primeiro método difuso (camadas não-dominadas) foi originalmente desenvolvido para resolver problemas de tomada de decisão com informação difusa. Esse método torna possível encontrar uma seqüência de camadas não-dominadas (núcleos) de alternativas em uma estrutura de preferências difusas, que não necessariamente representa uma ordem linear (total). Os núcleos subsequentes são grupos de alternativas que têm os maiores ranks dentre as alternativas que não pertencem aos núcleos prévios, de maior nível. O primeiro núcleo corresponde às alternativas de maiores ranks em todo o conjunto considerado.

O segundo método difuso (ranges) tenta constatar a credibilidade da declaração "a j-ésima alternativa se encontra exatamente na p-ésima posição na ordem por ranges". Os resultados são livres de ambigüidade no caso de uma relação de ordem linear (total) subjacente aos dados; caso contrário, um cuidado especial deve ser tomado para a interpretação dos resultados. O procedimento de otimização, desenvolvido para manusear o caso geral (normalizado ou não-normalizado), permite ao usuário decidir se normalizará a matriz relacional difusa antes do procedimento de ordenamento (ver opção NORM). Uma interpretação cuidadosa dos resultados é necessária depois da normalização. Usualmente dados incompletos resultam em uma matriz relacional não-normalizada, especialmente quando DATA=RAWC é usado e o número de alternativas selecionadas em respostas individuais é menor do que o número de alternativas possíveis. Apesar de uma matriz não-normalizada dar resultados cujos níveis de incerteza são maiores, ela pode fornecer uma visão mais realista sobre a relação latente determinando os dados; de fato, a normalização pode ser interpretada como um tipo de extrapolação.

Dois tipos de relações individuais de preferências (estrita ou fraca) podem ser especificadas, ambas no caso de dados representando uma seleção de alternativas, e no caso de dados representando um ordenamento de alternativas por ranges.

1. Dados representando uma **seleção** de alternativas.
   - **Preferências estritas:** cada alternativa selecionada é considerada ter um **único** (diferente) rango, enquanto às não selecionadas é dado o mesmo rank mais baixo.
   - **Preferências fracas:** todas as alternativas selecionadas são assumidas possuírem um rank **comum**, que é maior do que o rank das não selecionadas.
2. Dados representando um **ordenamento** de alternativas.
   - **Preferência estrita:** para todas as alternativas ordenadas faz-se a suposição que tenham valores **diferentes**, e relações entre alternativas possuindo o mesmo rank não são consideradas no cálculo da relação de preferências ao longo das alternativas.
   - **Preferência fraca:** alternativas com **mesmo** rank são consideradas durante o cálculo.

34.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar os casos dos dados de entrada, e o parâmetro VARS é usado para selecionar variáveis.

Transformando dados. Declarações de Recode podem ser usadas. Note que somente a parte inteira das variáveis recodificadas é usada pelo programa, i.e. variáveis recodificadas são arredondas para o inteiro mais próximo.

Ponderando dados. Dados podem ser ponderados por valores inteiros. Note que pesos com valores decimais são arredondados para o inteiro mais próximo. Quando o valor da variável de ponderação para um caso é zero, negativo, perdido ou não-numérico, então o caso é sempre evitado; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, serão usados para checar a existência de dados perdidos. Para DATA=RAWC, as variáveis com dados perdidos são evitadas, para DATA=RANKS, os valores de dados perdidos são substituídos pelo menor rango.

34.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Variáveis descritoras de registros, e registros-C, se houver, somente para variáveis usadas na execução.

Dados inválidos. Mensagens sobre dados incorretos (rejeitados).

Métodos baseados em lógica difusa (METHOD=NOND/RANKS)

Matriz de relação. Uma matriz quadrada representando uma relação difusa é impressa pelas linhas. Se as linhas possuem mais de dez elementos, elas são continuadas em linhas subseqüentes.

Descrição das relações. Depois de imprimir o tipo de relação, três medidas são dadas, as quais caracterizam concisamente a relação: coerência absoluta, índices de dominância absoluta e de intensidade.

Resultados de análise. Os resultados são apresentados em diferentes formas para cada método.

Para METHOD=NOND os núcleos são impressos seqüencialmente do rank mais alto para o mais baixo e, para cada um deles, é dada a seguinte informação:

seu número seqüencial, com o nível de certeza,
os códigos e labels de código das alternativas, ou os números de variáveis e nomes (até oito caracteres),
os valores da função de filiação das alternativas indicando com que intensidade elas estão conectadas ao núcleo; valores de filiação das alternativas pertencendo aos núcleos anteriores são substituídos por asteriscos,

lista de alternativas pertencendo ao núcleo com o maior valor de filiação (alternativas mais críveis).

Para METHOD=RANKS a matriz relacional normalizada é impressa primeiro se normalização é requisitada. Os resultados são então impressos em duas formas para facilidade de interpretação.

1. Todas as alternativas são listadas seqüencialmente com, para cada uma:
   o código e label de código da alternativa, ou o número de variável e nome,
   os valores de função de filiação da alternativa indicando quão forte ela está conectada a cada rango,
   a lista do(s) rango(es) mais críveis para aquela alternativa.
2. Todos os rangess são listados seqüencialmente com, para cada um:
   o número do rango,
   o código e label de código das alternativas, ou os números e nomes das variáveis,
   os valores de função de filiação da alternativa indicando quão forte ela está conectada a cada rango,
   a lista da(s) alternativa(s) mais críveis para aquele rango.

Método baseado em lógica clássica (METHOD=CLAS)

Resultados de análise. Para cada estrutura relacional de "dominância" final resultando de uma análise, as diferenças de ranks e as proporções populacionais de mínimo/máximo especificados pelo usuário são impressas, seguidos pela lista de núcleos sucessivos não-dominados (identificados pelos seus números seqüenciais) com as alternativas pertencendo a eles.

Nota. Alternativas recebem nomes que são os primeiros 8 caracteres da nome de variável para DATA=RANKS ou com a label de código de 8 caracteres (se registros-C estiverem presentes no dicionário) para DATA=RAWC.

34.4 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Toda as variáveis de análise devem possuir valores inteiros positivos. Note que as variáveis com valores decimais são arredondadas para o inteiro mais próximo.

Preferências são representadas de 2 maneiras nos dados. A seguinte ilustração mostra isso.

Suponha que dados devem ser coletados sobre as preferências que trabalhadores possuem em relação aos seus trabalhos:

Próprio escritório
Salário alto
Férias longas
Supervisão mínima
Colegas compatíveis

As duas maneiras de representar isso em um questionário são:

1. DATA=RAWC
   Nesse caso, os fatores são codificados (e.g. 1 a 5) e o respondente é induzido a pegá-los em ordem de preferência. As variáveis nos dados representariam o rango, e.g.

```
V6  Fator mais importante
V7  Segundo fator mais importante
.
.
V10 Fator menos importante
```

   e os códigos designados para cada uma dessas variáveis pelo respondente representariam os fatores (e.g. 1=próprio escritório, 2=salário alto, etc.).

Nem todos os possíveis fatores precisam ser selecionados, alguém pode perguntar 3 questões importantes, especificando apenas essas variáveis na lista de variáveis e.g. V6, V7, V8. O número de diferentes fatores sendo usados é especificado com o parâmetro NALT.

2. DATA=RANKS
   Aqui, cada fator é listado no questionário como uma variável, e.g.

```
V13 Próprio escritório
V14 Alto salário
.
.
V17 Coleguas compatíveis
```

   e o respondente é convidado a designar o rank de cada um, onde 1 é dado ao fator mais importante, 2 ao próximo mais importante, etc. Aqui as variáveis representam os fatores e seus valores representam o rango. Para cada variável deve ser designado um rank e todos os fatores sempre entrarão na análise. Os ranks devem ser codificados de 1 até n onde n é o número de variáveis sendo consideradas.

Notas.

1. Se DATA=RANKS, o código 0 e todos os códigos maiores do que n onde n é o número de variáveis (i.e. número de alternativas) são tratados com valores perdidos e são designados para o rank mais baixo.

2. Se DATA=RAWC, os primeiros códigos NALT diferentes encontrados ao ler-se os dados (excluindo 0) são usados como códigos válidos. Outros códigos encontrados mais tarde nos dados são tomados como códigos inválidos. Zero é sempre tratado como código ilegal. Se o número de alternativas selecionadas pelos respondentes for menor do que NALT, então as alternativas não selecionadas aparecem nos resultados com código zero e label de código vazia.

34.5 Estrutura de Setup

```
$RUN RANK

$FILES
     Especificações de arquivo

$RECODE (optional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de análise (repetida como requerida)
        (para lógica clássica apenas)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados

Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default  IDAMS.LST)
```

34.6 Declarações de Controle de Programa

Referir-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais detalhadas das declarações de controle de program, itens 1-4 abaixo

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V2=11
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  FIRST RUN OF RANK
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

   ```
   Exemplo:  DATA=RANKS  PREF=STRICT  MDVALUES=NONE  VARS=(V11-V13)
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

   MAXCASES=n
   O número máximo de casos (depois da filtragem) a serem usados como arquivo de entrada.
   Default: Todos os casos serão usados.

   MDVALUES=**BOTH**/MD1/MD2/NONE
   Quais valores de dados perdidos serão utilizados para as variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".
   Para DATA=RAWC, variáveis com dados perdidos não incluídas no ordenamento.
   Para DATA=RANKS, valores de dados perdidos são recodificados no rank mais baixo.

   VARS=(lista de variáveis)
   Uma lista de variáveis-V ou -R a ser usada no procedimento de ordenamento.
   Não há default.

   WEIGHT=número de variável
   O número da variável de ponderação, caso os dados sejam ponderados.

   METHOD=(**CLASSICAL**/NOCLASSICAL, NONDOMINATED, RANKS)
   Especifica o método a ser utilizado na análise.
   - CLAS Método de lógica clássica (ELECTRE).
   - NOND Método-1 difuso, chamado camadas não dominadas.
   - RANK Método-2 difuso, chamado ranges.

   DATA=**RAWC**/RANKS
   Tipo de dados.
   - RAWC As variáveis correspondem a ranks (a primeira variável na lista possui o primeiro rango, a segunda o segundo rango, etc.), enquanto seus valores são o número de código da alternativa selecionada.
   - RANK Variáveis representam alternativas, seus valores sendo ranks das alternativas correspondentes.

PREF=**STRICT**/WEAK

Determina o tipo de relação de preferência a ser usada na análise.

STRI Uma relação de preferências estrita é utilizada.

WEAK Uma relação de preferências fraca é utilizada.

NALT=**5**/n

(DATA=RAWC somente). O número total de alternativas a serem ordenadas.

Nota: Se DATA=RANKS, o número de alternativas é automaticamente ajustado para o número de variáveis de análise.

NORMALIZE=**NO**/YES

(METHOD=RANKS somente).

NO Sem normalização.

YES Normalização da matriz relacional é executada antes do cálculo do valor da função de filiação das alternativas.

PRINT=CDICT/DICT

CDIC Imprime o dicionário de entrada para variáveis acessadas com registros-C, se houver.

DICT Imprime o dicionário de entrada sem registros-C.

4. **Especificações de análise** (condicional: somente no caso do método de lógica clássica). As regras de codificação são as mesmas dos parâmetros. Cada especificação de análise deve se iniciar em uma nova linha.

```
Exemplo:  PCON=66   DDIS=4   PDIS=20
```

DCON=**1**/n

Diferença de ranks controlando a concordância em opiniões individuais (casos). Deve ser um inteiro no intervalo 0 até NALT-1.

PCON=**51**/n

Proporção mínima de concordância individual, expressa como uma percentagem, requerida na opinião coletiva. Deve ser um inteiro no intervalo 0 até 99. O valor default significa que pelo menos 51% de concordância é requerida para uma concordância coletiva.

DDIS=**2**/n

Diferença de ranks controlando a discordância em opiniões individuais (casos). Deve ser um inteiro no intervalo 0 a NALT-1.

PDIS=**10**/n

Proporção máxima de discordância individual, expressa como uma percentagem, tolerada na opinião coletiva. Deve ser um inteiro no intervalo o até 100. O valor default significa que não mais de 10% de discordância individual é tolerada.

34.7 Restrições

1. O número máximo de variáveis permitidas em qualquer execução é 200, incluindo aquelas usadas em declarações de Recode e a variável de ponderação.
2. O número máximo de variáveis de análise é 60.

34.8 Exemplos

Exemplo 1. Determinação da ordem de ranks de alternativas usando dados coletados na forma de ordenamento de alternativas; há 10 alternativas, é assumida uma relação de preferências fraca, e a análise deve ser feita utilizando-se o método Ranks.

```
$RUN RANK
$FILES
PRINT  = RANK1.LST
DICTIN = PREF.DIC                    arquivo Dicionário de entrada
DATAIN = PREF.DAT                    arquivo Dados de entrada
$SETUP
RANK - ORDERING OF ALTERNATIVES : RANKS METHOD
DATA=RANKS  PREF=WEAK  METH=(NOCL,RANKS)  VARS=(V21-V30)
```

Exemplo 2. Determinação da ordem de ranks de alternativas usando dados coletados na forma de uma seleção de prioridades; três alternativas são selecionadas de 20 e a ordem das variáveis determina a prioridade de seleção; relação de preferência estrita é assumida; ambos os métodos difusos são requisitados na análise.

```
$RUN RANK
$FILES
    como no Exemplo 1
$SETUP
RANK - ORDERING OF ALTERNATIVES : TWO FUZZY METHODS
NALT=20  METH=(NOCL,NOND,RANKS)  VARS=(V101-V103)
```

Exemplo 3. Determinação de uma ordem de ranks de alternativas usando dados coletados na forma de uma seleção de prioridades; 4 alternativas são selecionadas de 15 e a ordem das variáveis não determina a prioridade de seleção (preferência fraca); quatro análises de lógica clássica são executadas mantendo-se as diferenças de ranks sempre iguais a 1, mas aumentando a proporção de discordância e diminuindo a proporção de discordância.

```
$RUN RANK
$FILES
    como no Exemplo 1
$SETUP
RANK - ORDERING OF ALTERNATIVES : CLASSICAL LOGIC
PREF=WEAK  NALT=15  METH=CLAS  VARS=(V21,V23,V25,V27)
PCON=75  DDIS=1  PDIS=5
PCON=66  DDIS=1  PDIS=10
PCON=51  DDIS=1  PDIS=15
PCON=40  DDIS=1  PDIS=20
```

Capítulo 35

Diagramas de Dispersão (SCAT)

35.1 Descrição Geral

SCAT é um programa de análise bivariada que produz diagramas de dispersão, estatísticas univariadas, e estatísticas bivariadas. Os diagramas de dispersão são plotados em um sistema de coordenadas retangulares; para cada combinação de valores de coordenadas que aparece nos dados, a freqüência da sua ocorrência é mostrada.

SCAT é útil para mostrar relações bivariadas se os números de valores diferentes para cada variável é grande e o número de casos de dados contendo qualquer um dos valores é pequeno. Se, contundo, uma variável assume relativamente poucos valores diferentes em um grande número de casos de dados, o programa TABLES é mais apropriado.

Formato da plotagem. Cada plotagem desejada é definida separadamente ao se especificar as duas variáveis a serem usadas (chamadas variáveis X e Y). As escalas dos eixos são ajustadas separadamente para cada plotagem para permitir que variáveis com escalas radicalmente diferentes possam ser plotadas em relação a elas mesmas, sem perda de discriminação. Normalmente, o programa plota a variável com o maior intervalo (antes do re-escalonamento) ao longo do eixo horizontal. No entanto, o usuário pode requerer que a variável X seja sempre plotada ao longo do eixo horizontal. As freqüências reais são introduzidas no diagrama se eles forem menores que 10. Para freqüências de 10-65, as letras do alfabeto são utilizadas. Se a freqüência de um ponto é maior que 65, um asterisco é colocado no diagrama. Esse esquema de codificação é parte dos resultados para facilidade de referência.

Estatísticas. A média, desvio-padrão, valores mínimo e máximo são impressos para cada variável acessada, incluindo o filtro de plotagem e variável de ponderação, se houver. Para cada plotagem o programa também imprime a média, desvio-padrão, contagem de casos e intervalo para as duas variáveis, coeficiente de correlação r de Pearson, a constante de regressão, e o coeficiente de regressão não padronizado para prever Y partindo de X.

35.2 Características Padrão de IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar subconjuntos de casos dos dados de entrada. Além disso, uma variável de filtro de plotagem e intervalo de valores podem ser especificados para restringir os casos de dados incluídos em uma plotagem particular. As variáveis a serem plotadas são especificadas em pares com parâmetros de plotagem.

Transformando dados. Declarações de Recode podem ser usadas. Note que para variáveis-R, o número de decimais a ser retido é especificado pelo parâmetro NDEC.

Ponderando dados. Uma variável de ponderação pode ser especificada para cada plotagem. Ambas, variáveis-R e -V, com casas decimais são multiplicadas por um fator de escala para obter-se valores inteiros. Ver a seção “Dataset de Entrada” abaixo.

Quando o valor da variável de ponderação para um caso for zero, negativa, perdida ou não-numérica, então o caso é sempre evitado; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser usados para checar a existência de dados perdidos. As estatísticas univariadas que aparecem no início dos resultados, imediatamente seguindo o dicionário, são baseadas em todos os casos que possuem dados válidos em cada variável considerada separadamente. Para as plotagens, o programa elimina casos que possuam dados perdidos em uma ou ambas as variáveis em um a plotagem em particular. Essa deleção por pares afeta também as estatísticas univariadas e bivariadas que são impressas no topo de cada plotagem.

35.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, somente para variáveis usadas na execução.

Estatísticas univariadas. Os seguintes são impressos para cada variável referenciada, incluindo filtro de plotagem e variáveis de ponderação: valores mínimos e máximos, média e desvio-padrão, e o número de casos com valores de dados válidos.

Chave para o esquema de codificação usada nas plotagens. Uma tabela mostrando a correspondência entre as freqüências atuais e os códigos usados nas plotagens.

Plotagem e estatísticas. Para cada plotagem requisitada, um diagrama de dispersão de 8 1/2 polegadas por 12 polegadas é impresso. Estatísticas univariadas (médias, desvios-padrões) e estatísticas bivariadas (r de Pearson, a constante de regressão A, e o coeficiente não-padronizado da regressão B) são impressos no topo da plotagem.

35.4 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis de análise e de filtro de plotagem devem ser numéricas; elas devem ter valores inteiros ou decimais. Variáveis com decimais são multiplicadas por fator de escala para obter-se valores inteiros. Esse fator é calculado como 10^n onde n é o número de decimais obtido do dicionário para variáveis-V e do parâmetro NDEC para variáveis-R; ele é impresso para cada variável.

35.5 Estrutura de Setup

```
$RUN SCAT

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de plotagem (repetidas como requisitadas)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados

Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT for usado)
DATAxxxx   dados de entrada (omitir se $DATA for usado)
PRINT      resultados (default  IDAMS.LST)
```

35.6 Declarações de Controle de Programa

Reporte-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-4 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V21=6 AND V37=5
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  STUDY 600. JULY 16, 1999. AGE BY HEIGHT FOR SUBSAMPLE 3
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa. Novos parâmetros são precedidos por um asterisco.

   ```
   Exemplo:  BADD=MD2
   ```

   INFILE=**IN**/xxxx
   Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
   Default ddnames: DICTIN, DATAIN.

   BADDATA=**STOP**/SKIP/MD1/MD2
   Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

   MAXCASES=n
   O número máximo de casos (depois da filtragem) a serem usados do arquivo de entrada.
   Default: Todos os casos serão utilizados.

MDVALUES=**BOTH**/MD1/MD2/NONE
Quais valores de dados perdidos devem ser utilizados para as variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".

* NDEC=**0**/n
Número de decimais (máximo 4) a ser retido para variáveis-R.

PRINT=CDICT/DICT
CDIC Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT Imprime o dicionário sem registros-C.

4. **Especificações de plotagem.** Um conjunto para cada plotagem. As regras de codificação são as mesmas das dos parâmetros. Cada especificação de plotagem deve se iniciar em uma nova linha.

```
Exemplo:  X=V3   Y=R17   FILTER=(V3,1,1)
```

X=número de variável
Número de variável da variável X.

Y=número de variável
Número de variável da variável Y.

WEIGHT=número de variável
O número da variável de ponderação, se os dados forem poderados.

FILTER=(número de variável, código válido mínimo, código máximo válido)
Filtro de plotagem. Apenas aqueles casos onde o valor da variável de filtro for maior ou igual ao código mínimo, e menor ou igual ao código máximo, serão introduzidos na plotagem. Por exemplo, para especificar que apenas os casos com códigos 0-40 na variável 6 devem ser incluídos, especifique: FILTER=(V6,0,40).

HORIZAXIS=**MAXRANGE**/X
MAXR Plota a variável com o maior intervalo ao longo do eixo horizontal.
X Plota sempre a variável X ao longo do eixo horizontal.

35.7 Restrições

1. Não mais do que 50 variáveis podem ser usadas em uma execução do programa. Esse máximo inclui tudo: variáveis X e Y, variáveis de filtro de plotagem, pesos e variáveis utilizadas em declarações de Recode.
2. Não há limite no número de plotagens, mas SCAT produz apenas 5 plotagens para cada passagem de dados de entrada.

35.8 Exemplo

Geração de duas plotagens (ponderada pela variável V100 e não-ponderada) repetidas para 3 subconjuntos de dados diferentes.

```
$RUN SCAT
$FILES
PRINT  = SCAT1.LST
DICTIN = MY.DIC                          arquivo Dicionário de entrada
DATAIN = MY.DAT                          arquivo Dados de entrada
$SETUP
GENERATION OF TWO PLOTS REPEATED FOR EACH SUBSET OF DATA
*                                (valors default obtidos para todos parâmetros)
X=V21  Y=V3  FILTER=(V5,1,2)
X=V21  Y=V3  FILTER=(V5,1,2)  WEIGHT=V100
X=V21  Y=V3  FILTER=(V5,3,3)
X=V21  Y=V3  FILTER=(V5,3,3)  WEIGHT=V100
X=V21  Y=V3  FILTER=(V5,4,7)
X=V21  Y=V3  FILTER=(V5,4,7)  WEIGHT=V100
```

Capítulo 36

Busca de Estrutura (SEARCH)

36.1 Descrição Geral

SEARCH é um procedimento de segmentação binário usado para desenvolver um modelo preditivo para variáveis dependentes. Ele busca ao longo de um conjunto de variáveis preditoras, aqueles preditores que mais aumentam a habilidade do pesquisador em explicar a variância ou a distribuição de uma variável dependente. A questão "que partição dicotômica, e em que variável preditora, nos dará o maior aumento em nossa habilidade de predizer os valores da variável dependente?", inserida em um esquema iterativo, é a base para o algoritmo usado nesse programa.

SEARCH divide a amostra, por meio de uma série de partições binárias, em séries de subgrupos mutuamente exclusivos. Os subgrupos são escolhidos tal que, a cada passo no procedimento, a partição em dois novos subgrupos explica mais a variância ou distribuição (reduz o erro preditivo um pouco mais) do que a partição em qualquer outro par de subgrupos.

SEARCH pode executar as seguintes funções:

* Maximizar diferenças em médias de grupos, linhas de regressão de grupo, ou distribuições (critério chi-quadrado de máxima verossimilhança).
* Ranquear os preditores para dar a eles preferência na partição.
* Sacrificar potência explanatória por simetria.
* Iniciar depois que uma estrutura de árvore parcial especificada tiver sido gerada.

Gerando um dataset de resíduos. Resíduos podem ser computados e produzidos como um arquivo de dados descrito por um dicionário do IDAMS. Ver a seção "Dataset de Resíduos de Saída" para detalhes a respeito desse conteúdo.

36.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. As variáveis dependentes são especificadas no parâmetro DEPVAR, e os preditores são especificados no parâmetro VARS nos declarações de preditores.

Transformando dados. Declarações de Recode podem ser usadas.

Ponderando dados. Uma variável pode ser usada para ponderação dos dados de entrada; essa variável de ponderação pode conter valores inteiros ou decimais. Quando o valor da variável de ponderação para o caso for zero, negativo, perdido ou não-numérico, então o caso é evitado; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos. Casos com dados perdidos em uma variável dependente contínua ou covariate são deletados automaticamente. Casos com dados perdidos em uma variável dependente categórica podem ser excluídos usando-se uma declaração de filtro ou especificando-se códigos válidos com o parâmetro DEPVAR. Casos com dados perdidos nas variáveis preditoras não são automaticamente excluídos. Contudo, a declaração de filtro e/ou o parâmetro CODES pode ser usado para esse propósito.

36.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, somente para variáveis usadas na execução.

Outliers. (Opcional: ver o parâmetro PRINT). Outliers com os valores de variável de ID e os valores da variável dependente.

Traço. (Opcional: ver as opções dos parâmetros PRINT, TRACE e FULLTRACE). O traço das partições para cada preditor para cada partição contendo: os grupos candidatos à partição, todas as partições elegíveis para cada preditor, a melhor partição para cada preditor e o grupo "split-on".

Resumo de análise contendo as distribuições de análise de variância, o resumo da partição e o resumo dos grupos finais.

Tabelas de resumo de preditores. (Opcional: ver as opções dos parâmetros PRINT, TABLE, FIRST e FINAL). As tablas de grupo primeiro (PRINT=FIRST), as tabelas de grupos finais (PRINT=FINAL) ou as tabelas de todos grupos (PRINT=TABLE) contendo um resumo das melhores partições para cada preditor para cada grupo. As tabelas são impressas em ordem de grupo reversa, i.e. o último grupo vem primeiro.

Diagrama de árvore. (Opcional: Opcional: ver o parâmetro PRINT). Diagrama de árvore hierárquico. Cada nó (caixa) da árvore contém: número do grupo, número de casos (N), número da partição, número da variável preditora, média da variável dependente (para análise de médias), e média da variável dependente e covariate, e inclinação (para análise de regressão)

36.4 Dataset de Resíduos de Saída

Resíduos podem opcionalmente ser produzidos na forma de um arquivo de dados descrito por um dicionário do IDAMS. (Ver o parâmetro WRITE). Para médias e análise de regressão, e análise de chi-quadrado com múltiplas variáveis dependentes, cada registro de saída contém: uma variável de ID, a variável de grupo, variáveis dependentes, variáveis dependentes (calculadas) preditas, resíduos, e um peso, se houver.

Para análise de chi-quadrado com uma variável dependente categórica, ele contém: uma variável de ID, a variável de grupo, a primeira categoria da variável dependente, a primeira categoria predita (calculada) da variável dependente, o resíduo para a primeira categoria da variável dependente, a segunda categoria da variável dependente, a segunda categoria predita (calculada) da variável dependente, o resíduo para a segunda categoria da variável dependente, etc., e um peso, se houver.

As características das variáveis de saída são as seguintes:

	No. de variável	Nome	Largura de campo	No. de decimais	Código MD1
(variável de ID)	1	mesmo do entrada	*	0	mesmo do entrada
(variável de grupo)	2	Group variable	3	0	999
(dependente var 1)	3	mesmo do entrada	*	**	mesmo do entrada
(predita var 1)	4	mesmo do entrada cal	7	***	9999999
(resíduo para var 1)	5	mesmo do entrada res	7	***	9999999
(dependente var 2)	6	mesmo do entrada	*	**	mesmo do entrada
(predito var 2)	7	mesmo do entrada cal	7	***	9999999
(resíduo para var 2)	8	mesmo do entrada res	7	***	9999999
...	.	...	.	...	...
(peso-se ponderado)	n	mesmo do entrada	*	**	mesmo do entrada

*	transferido do dicionário de entrada para variáveis-V ou 7 para variáveis R
**	transferido do dicionário de entrada para variáveis-V ou 2 para variáveis R
***	6 mais o no. de decimais para a variável dependente menos a largura da variável dependente; se isso for negativo, então 0.

Se o valor calculado ou resíduo exceder a largura de campo alocada, será substituído pelo código MD1.

36.5 Dataset de Entrada

A entrada é um arquivo de dados descrito por um dicionário do IDAMS. Todas as variáveis usadas na análise devem ser numéricas; elas podem conter valores inteiros ou decimais. A variável dependente pode ser contínuo ou categórica. Variáveis preditoras podem ser ordinais ou categóricas. A variável de ID de caso pode ser alfabética.

36.6 Estrutura de Setup

```
$RUN SEARCH

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de preditor
     5. Especificações de partição pré-definida (opcional)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de resíduos de saída
DATAyyyy   dados de resíduos de saída
PRINT      resultados (default  IDAMS.LST)
```

36.7 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-5 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V3=5
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

```
Exemplo:  SEARCHING FOR STRUCTURE
```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

```
Exemplo:  DEPV=V5
```

INFILE=**IN**/xxxx
Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada.
Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2
Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
O número máximo de casos (depois da filtragem) a serem usados do arquivo de entrada.
Default: Todos os casos serão usados.

ANALYSIS=**MEAN**/REGRESSION/CHI

MEAN	Análise de médias.
REGR	Análise de regressão.
CHI	Análise de chi-quadrado. Com uma única variável dependente, a lista default de códigos 0-9 será utilizada e nenhuma verificação de dados perdidos será feita.

DEPVAR=número de variável/(lista de variáveis)
A variável dependente ou variáveis. Note que a lista de variáveis somente pode ser fornecida quando ANALYSIS=CHI é especificado.
Não há default.

CODES=(lista de códigos)
Uma lista de códigos somente será fornecida para ANALYSIS=CHI e uma variável dependente. Note que nesse caso nenhuma verificação de dados perdidos será feita para a variável dependente e somente casos com códigos listados são usados na análise.

COVAR=número de variável
O número de variável do covariate. Deve ser fornecido para ANALYSIS=REGR.

WEIGHT=número de variável
O número da variável de ponderação se os dados devem ser ponderados.

MINCASES=**25**/n
Número mínimo de casos em um grupo.

MAXPARTITIONS=**25**/n
Número máximo de partições.

SYMMETRY=**0**/n
A porção de potência explanatória que se deseja perder para obter simetria, expressa em percentual.

EXPL=**0.8**/n
Aumento mínimo em potência explanatória requerido para uma partição, expresso em percentual.

OUTDISTANCE=**5**/n
Número de desvios-padrões em relação a média do grupo-parente definindo um outlier. Note que outliers são reportados se PRINT=OUTL for especificado, mas eles não são excluídos da análise.

IDVAR=número de variável
Variável a ser produzida com os resíduos e/ou impressa com cada caso classificado como outlier.

WRITE=RESIDUALS/CALCULATED/BOTH
Resíduos e/ou valores calculados devem ser escritos como um dataset de IDAMS.

RESI	Produz os valores de resíduos apenas.
CALC	Produz os valores calculados apenas.
BOTH	Produz ambos, resíduos e valores calculados.

OUTFILE=**OUT**/yyyy
Aplicável somente se WRITE é especificado.
Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de resíduos de saída.
Default ddnames: DICTOUT, DATAOUT.

PRINT=(CDICT/DICT, TRACE, FULLTRACE, TABLE, FIRST, FINAL, TREE, OUTLIERS)

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
TRAC	Imprime o traço das partições para cada preditor para cada partição.
FULL	Imprime o traço completo de partições de cada preditor, incluindo partições elegíveis, porém, sub-ótimas.
TABL	Imprime as tabelas de resumo de preditor para todos os grupos.
FIRS	Imprime as tabelas de resumo de preditor para o primeiro grupo.
FINA	Imprime as tabelas de resumo de preditor para os grupos finais.
TREE	Imprime o diagrama de árvore hierárquica.
OUTL	Imprime os outliers com variável de ID e valores de variável dependente.

4. **Especificações de preditor** (mandatório). Fornece um conjunto de parâmetros para cada grupo de preditores que podem ser descritos com os mesmos valores de parâmetro. As regras de codificação são as mesma das dos parâmetros. Cada especificação de preditor deve começar em uma nova linha.

```
Exemplo:  VARS=(V8,V9) TYPE=F
```

VARS=(lista de variáveis)
Variáveis preditoras nas quais os outros parâmetros se aplicam.
Não há default.

TYPE=**M**/F/S
A restrição do preditor.

M	Preditores são considerados "monotônicos", i.e. os códigos dos preditores devem ser mantidos adjacentes durante o scan de partição.
F	Códigos de preditores são considerados "livre".
S	Códigos de preditores serão "selecionados" e separados dos códigos remanescentes ao se formarem partições de teste.

CODES=**(0-9)**/maxcode/(lista de códigos)
Ou o valor do código mais aceitável ou uma lista de códigos aceitáveis. Os códigos podem ir de 0 a 31. Casos com códigos fora do intervalo 0 a 31 são sempre descartados.

RANK=n
Rango designado. Se é desejado o ordenamento, designe um rango de preditor de 0 a 9. Um rango zero indica que estatísticas devem ser computadas para os preditores, mas eles não devem ser usados na partição.

5. **Especificações de partição pré-definida** (opcional). Se partições predefinidas são desejadas, forneça um conjunto de parâmetros para cada partição pré-definida. As regras de codificação são as mesmas das dos parâmetros. Cada especificação de partição pré-definida deve se iniciar em uma nova linha.

```
Exemplo:  GNUM=1  VAR=V18  CODES=(1-3)
```

GNUM=n
Número do grupo para partição. Grupos são especificados em ordem crescente, onde a amostra original completa é o grupo 1. Cada conjunto de parâmetros forma dois novos grupos.
Não há default.

VAR=número de variável
Variável preditora usada para fazer a partiçaõ.
Não há default.

CODES=(lista de códigos)
Lista dos códigos preditores definindo o primeiro subgrupo. Todos os outros códigos pertencerão ao segundo subgrupo.
Não há default.

36.8 Restrições

1. Número mínimo de casos requeridos é 2 * MINCASES.
2. Número máximo de preditores é 100.
3. Valor de preditor máximo é 31.
4. Número máximo de código de variáveis categóricas é 400.
5. Número máximo de partições pré-definidas é 49.
6. Se a variável de ID é alfabética com largura > 4, somente os quatro primeiros caracteres são usados.

36.9 Exemplos

Exemplo 1. Análise de médias com cinco variáveis preditoras; mínimo de 10 casos por grupo são requisitados; outliers de mais de 3 desvios-padrões em relação á média do grupo-parente são reportados; casos são identificados pela variável V1.

```
$RUN SEARCH
$FILES
PRINT   = SEARCH1.LST
DICTIN  = STUDY.DIC                    arquivo Dictionário de entrada
DATAIN  = STUDY.DAT                    arquivo Dados de entrada
$SETUP
MEANS ANALYSIS  -  FIVE PREDICTOR VARIABLES
DEPV=V4  MINC=10  OUTD=3  IDVAR=V1  PRINT=(TRACE,TREE,OUTL)
VARS=(V3-V5,V12)
VARS=V21  TYPE=F  CODES=(1-4)
```

Exemplo 2. Análise de regressão com seis variáveis preditoras; valores de resíduos e calculados devem ser computados e salvos em um dataset (casos são identificados pela variável V2).

```
$RUN SEARCH
$FILES
PRINT   = SEARCH2.LST
DICTIN  = STUDY.DIC                    arquivo Dicionário de entrada
DATAIN  = STUDY.DAT                    arquivo Dados de entrada
DICTOUT = RESID.DIC                    arquivo Dicionário para resíduos
DATAOUT = RESID.DAT                    arquivo Dados para resíduos
$SETUP
REGRESSION ANALYSIS  -  SIX PREDICTOR VARIABLES
ANAL=REGR  DEPV=V12  COVAR=V7  MINC=10  IDVAR=V2  -
     WRITE=BOTH  PRINT=(TRACE,TABLE,TREE)
VARS=(V3-V5,V18)
VARS=V22  TYPE=F
```

Exemplo 3. Análise de chi-quadrado com uma variável categórica dependente e códigos selecionados; as duas primeiras partições estão pré-definidas.

```
$RUN SEARCH
$FILES
DICTIN = STUDY.DIC                     arquivo Dicionário de entrada
DATAIN = STUDY.DAT                     arquivo Dados de inpu
$SETUP
CHI ANALYSIS - ONE DEPENDENT CATEGORICAL VARIABLE, PREDEFINED SPLITS
ANAL=CHI  DEPV=V101  CODES=(1-5)  MINC=5  PRINT=(FINAL,TREE)
VARS=(V3,V8)  TYPE=S
GNUM=1  VAR=V8  CODES=3
GNUM=2  VAR=V3  CODES=(1,2)
```

Capítulo 37

Tabelas Univariadas e Bivariadas (TABLES)

37.1 Descrição Geral

O principal uso de TABLES é obter distribuições de freqüências univariadas e bivariadas com percentuais opcionais de linha, coluna e canto e estatísticas opcionais univariadas e bivariadas. Tabelas de valores médios podem também ser obtidas.

Tanto tabelas univariadas/bivariadas quanto estatísticas bivariadas podem ser produzidas em um arquivo de maneira que elas podem ser usadas com um programa gerador de relatórios, ou podem servir de entrada para GraphID ou outros pacotes como EXCEL para display gráfico.

Tabelas univariadas. Tanto freqüências univariadas quanto freqüências univariadas cumulativas podem ser geradas por um número qualquer de variáveis de entrada e podem também ser expressas como os percentuais ponderados e não ponderados da freqüência total. Além disso, a média de uma variável de célula pode ser obtida.

Tabelas bivariadas. Qualquer número de tabelas bivariadas pode ser gerado. Além das freqüências ponderadas/não-ponedaradas, uma tabela pode conter freqüências expressas como percentuais baseados nas marginais da linha, marginais da coluna ou total da tabela, e na média de uma variável de célula. Esses vários itens podem ser colocados em uma única tabela com, possivelmente, seis itens por célula, ou cada um pode ser obtido como uma tabela distinta.

Estatísticas univariadas. Para análises univariadas, as seguintes estatísticas estão disponíveis: média, moda, mediana, variância (não-viesada), desvio-padrão, coeficiente de variação, assimetria e curtose. Uma opção de quantile (NTILE) está também disponível. Divisão de três a dez partes pode ser requerida.

Estatísticas bivariadas. Para análises bivariadas, as seguintes estatísticas podem ser requeridas:

- testes-t de médias (assume populações independentes) entre pares de linhas,
- chi-quadrado, coeficiente de contigência e V de Cramer,
- Gama, Lambdas e Taus de Kendall,
- S (numerador da estatística de tau e de gamma), seus desvios padrão e normal, e sua variância,
- ro de Spearman,
- Estatísticas de Medicina Baseada em Evidência (EBM),
- testes não paramétricos: Wilcoxon, Mann-Whitney e Fisher.

Matrizes de estatísticas. Matrizes de quaisquer das estatísticas bivariadas exceto testes, estatísticas EBM ou estatísticas de S podem ser impressas ou escritas em um arquivo. Matrizes correspondentes de n's ponderadas e/ou não-ponderadas podem ser produzidas.

Tabelas de 3- e 4-way. Elas podem ser construídas fazendo-se uso das características de repetição e construção de subconjuntos. A variável de repetição pode ser vista como uma variável de controle ou painel. A característica de construção de subconjuntos pode ser usada para selecionar ainda mais casos para um grupo particular de tabelas.

Tabelas de somas. Tabelas cujas células contêm a soma de uma variável dependente podem ser construídas ao especificar-se a variável dependente como um peso. E.g. especifique WEIGHT=V208, onde V208 representa a renda dos respondentes, para se obter a renda total de todos os respondentes que pertencem a uma célula.

Nota. As seguintes opções estão disponíveis para controlar a aparência dos resultados:

Um título pode ser especificado para cada conjunto de tabelas.

Percentuais e valores médios, se requeridos, podem ser impressos em tabelas separadas.

O grid pode ser suprimido.

Linhas que não possuem entradas em seções particulares de uma grande tabela de freqüências podem ser impressas; tabelas com mais do que dez colunas são impressas em seções e o uso dessa opção de "linhas de zeros" assegura que as várias seções possuem o mesmo número de linhas (o que é importante se elas forem "cortadas" e "passadas" juntas).

37.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. Além disso, os filtros locais e fatores de repetição (chamados especificações de subconjunto) podem ser usados para selecionar um subconjunto de casos de uma tabela em particular. Para tabelas que são especificadas individualmente, as variáveis a serem usadas na tabela são selecionadas com os parâmetros de especificação de tabela R e C. Para conjuntos de tabelas, variáveis são selecionadas com os parâmetros de especificação ROWVARS e COLVARS.

Transformando dados. Declarações de Recode podem ser usadas. Note que para variáveis-R, o número de decimais a ser retido é especificado pelo parâmetro NDEC.

Ponderando dados. Uma variável de ponderação pode opcionalmente ser especificada para cada conjunto de tabelas. Ambas, variáveis-R e -V, com casas decimais são multiplicadas por um fator de escala para obter-se valores inteiros. Ver a seção "Dataset de Entrada" abaixo. Quando o valor da variável de ponderação para um caso for zero, negativa, perdida ou não-numérica, então o caso é sempre evitado; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos.

1. O parâmetro MDVALUES está disponível para indicar que valores de dados perdidos, se houver, devem ser usados para checar a existência de dados perdidos.
2. Freqüências univariadas e bivariadas são sempre impressas para todos os códigos de dados, quer eles representem ou não dados perdidos. Para remover completamente dados perdidos das tabelas, um filtro ou um subconjunto pode ser especificado. Alternativamente, valores máximo e/ou mínimo da variável de linha e de coluna podem ser definidos.
3. Casos com dados perdidos podem ser, opcionalmente, incluídos no cômputo das percentagens e estatísticas bivariadas. Isso pode ser feito utilizando-se o parâmetro de tabela MDHANDLING.
4. Casos com dados perdidos em uma variável de célula são sempre excluídos das tabelas univariadas e bivariadas.
5. Casos com dados perdidos são sempre excluídos do cômputo das estatísticas univariadas.

37.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, somente para variáveis usadas nessa execução.

Uma tabela de conteúdos para os resultados. Os conteúdos mostram cada tabela produzida e dá o número da página onde está localizada. As seguintes informações são fornecidas:

- números de variável de linha e de coluna (0 se não houver)
- número da variável para o valor médio - variável da célula (0 se não houver)
- número da variável de ponderação (0 se não houver)
- valores mínimo e máximo da linha (0 se não houver)
- valores mínimo e máximo da coluna (0 se não houver)
- nome do filtro e nome do fator de repetição
- percentuais: linha, coluna e total (T=requerido, F=não requerido)
- RMD: dados perdidos da variável-linha (T=delete, F=não delete)
- CMD: dados perdidos da variável-coluna (T=delete, F=não delete)
- CHI: chi-quadrado (T=requerido, F=não requerido)
- TAU: tau a, b ou c (T=requerido, F=não requerido)
- GAM: gamma (T=requerido, F=não requerido)
- TEE: testes-t (T=requerido, F=não requerido)
- EXA: teste não paramétrico de Fisher (T=requerido, F=não requerido)
- WIL: teste não paramétrico de Wilcoxon (T=requerido, F=não requerido)
- MW: teste não paramétrico de Mann-Whitney (T=requerido, F=não requerido)
- SPM: ro de Spearman (T=requerido, F=não requerido)
- EBM: estatísiticas de Medicina Baseada em Evidência (T=requerido, F=não requerido).

Tabelas que são requisitadas usando-se os parâmetros PRINT=MATRIX ou WRITE=MATRIX não são listadas nos contenúdos e são sempre impressas primeiro com números de página e tabela negativos.

Outras tabelas são impressas na ordem das especificações de tabela, exceto para tabelas onde são requisitadas apenas estatísticas univariadas, essas são sempre agrupadas e impressas no final.

Tabelas bivariadas. Cada tabela bivariada se inicia em uma nova página; uma tabela grande pode precisar de mais de uma página. Tabelas são impressas com até 10 colunas e 16 linhas por página dependendo do número de itens em cada célula. Colunas e linha são impressas para códigos que aperecem realmente nos dados. Linha e coluna de totais, e freqüências cumulativas marginais e percentuais, se requisitados, são impressos ao redor da borda da tabela.

Uma grande tabela é impressa em linhas verticais. Por exemplo, uma tabela com 40 códigos de linha e 40 códigos de coluna seria normalmente impressa em 12 páginas como indicado pelo diagrama seguinte, onde os números nas células mostram a ordem a qual as páginas são impressas:

		1º 10	2º 10	3º 10	4º 10	códigos
1º	16 códigos	1	4	7	10	
2º	16 códigos	2	5	8	11	
últimos 8 códigos		3	6	9	12	

Estatísticas bivariadas. (Opcional: ver o parâmetro de tabela STATS).

Testes-t. (Opcional: ver o parâmetro de tabela STATS). Se os testes-t foram requisitados, eles e suas médias e desvios-padrões da variável de coluna para cada linha são impressos em uma página separada.

Matrizes de estatísticas bivariadas. (Opcional: ver o parâmetro de tabela PRINT). O canto inferior esquerdo da matriz é impresso. Oito colunas e 25 linhas são impressas por página.

Matriz de N's. (Opcional: ver o parâmetro de tabela PRINT). Isso é impresso no mesmo formato da matriz correspondente.

Tabelas univariadas. (Opcional: ver o parâmetro de tabela CELLS). Normalmente cada tabela univariada é impressa começando em uma nova página. Freqüências, percentuais e valores médios de uma variável, se requisitados, para dez códigos, são impressos ao longo da página.

Estatísticas univariadas. (Opcional: ver o parâmetro de tabela USTATS).

Quantiles. (Opcional: ver o parâmetro de tabela NTILE). N-1 pontos são impressos; e.g. se quantiles são requisitados, o parâmetro NTILE é igualado a 4 e 3 três pontos de quebra serão impressos.

Números de página. Esses são da seguinte forma: ttt.rr.ppp onde

ttt = número da tabela
rr = número de repetição (00 se nenhuma repetição for usada)
ppp = número de página dentro da tabela.

37.4 Tabelas Univariadas/Bivariadas de Saída

Tabelas univariadas e/ou bivariadas com estatísticas requisitadas no parâmetro de tabela CELLS podem ser produzidas em um arquivo pela especificação de WRITE=TABLES. As tabelas estão no formato de matriz retangular do IDAMS (ver o capítulo "Dados em IDAMS"). Uma matriz é produzida para cada estatística requisitada. Se um fator de repetição é usado, uma matriz é produzida para cada repetição.

Colunas 21-80 no registro descritor de matriz contém descrição adicional da matriz da seguinte maneira:

21-40	Nome da variável de linha (para tabelas bivariadas)
41-60	Nome da variável de coluna.
61-80	Descrição dos valores da matriz.

Registros de identificação de variável (#R e #C) contêm valores de código e labels de código para a variável de linha e de coluna, respectivamente.

As estatísticas são escritas como registros de 80 caracteres de acordo com o formato de Fortran 7F10.2. Colunas 73-80 contém uma ID da seguinte maneira:

73-76	Identificação da estatística: FREQ, UNFR, ROWP, COLP, TOTP ou MEAN.
77-80	Número da tabela.

Note que os códigos de dados perdidos não estão incluídos na matriz.

37.5 Matrizes de Estatísticas Bivariadas de Saída

Estatísticas selecionadas podem ser produzidas em um arquivo. Se, por exemplo, gama e tau b foram selecionados, uma matriz de gama e uma matriz separada de tau b seria gerada. Matrizes de estatísticas bivariadas de saída são requisitados ao se especificar WRITE=MATRIX e os parâmetros de tabela ROWVARS ou ROWVARS e COLVARS. Se um fator de repetição é usado, uma matriz é produzida para cada repetição. As matrizes estão no formato de matrizes retangulares ou quadradas (ver o capítulo "Dados em IDAMS"). Os valores na matriz são escritos no formato de Fortran 6F11.5. Colunas 73-80 contém uma ID da seguinte maneira:

73-76	Identificação da estatística: TAUA, TAUB, TAUC, GAMM, LSYM, LRD, LCD, CHI, CRMV ou RHO.
77-80	Número da tabela.

Nota. Se somente ROWVARS é fornecido, registros de médias dummy e de desvios-padrões são escritos, 2 registros por 60 variáveis. O segundo formato de registro (#F) no dicionário especifica um formato de 60I1 para esses registros dummy. Isso é para que a matriz se conforme ao formato de uma matriz quadrada do IDAMS.

37.6 Dataset de Entrada

A entrada é um arquivo de dados descrito por um dicionário do IDAMS. Com a exceção de variáveis usadas no filtro principal, todas as outras variáveis usadas devem ser numéricas.

Nas distribuições e ponderações, variáveis (tanto V quanto R) com casas decimais são multiplicadas por um fator de escala para obter-se valores inteiros. O fator de escala é calculado como 10^n onde n é o número de decimais obtido do dicionário para variáveis-V e do parâmetro NDEC para variáveis-R; ele é impresso para cada variável.

Estatísticas univariadas sem distribuições são calculadas usando-se o número de casas decimais especificado no dicionário para variáveis-V e obtido do parâmetro NDEC para variáveis-R.

Campos contendo caracteres não-numéricos (incluindo campos em branco) podem ser tabulados ao se definir o parâmetro BADDATA como MD1 ou MD2. Ver o capítulo "O Arquivo Setup do IDAMS".

37.7 Estrutura de Setup

```
$RUN TABLES

$FILES
     Especificações de arquivo

$RECODE (opcional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros
     4. Especificações de subconjunto (opcional)
     5. TABLES
     6. Especificações de tabela (repetidas como requisitadas)

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados


Arquivos:
FT02       tabelas/matrizes de saída
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
PRINT      resultados (default  IDAMS.LST)
```

37.8 Declarações de Controle de Programa

Reportar-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais aprofundadas das declarações de controle de programa, itens 1-3 e 6 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo:  INCLUDE V3=6
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para nomear os resultados.

   ```
   Exemplo:  FREQUENCY TABLES
   ```

3. **Parâmetros** (mandatório). Para selecionar opções de programa. Novos parâmetros são precedidos por um asterisco.

   ```
   Exemplo:  BADDATA=SKIP
   ```

INFILE=**IN**/xxxx
: Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de entrada. Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2
: Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
: O número máximo de casos (depois da filtragem) a serem usados com arquivo de entrada. Default: Todos os casos serão utilizados.

MDVALUES=**BOTH**/MD1/MD2/NONE
: Quais valores de dados perdidos devem ser utilizados para as variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".

* NDEC=**0**/n
: Número de decimais (máximo 4) a ser retido para variáveis-R.

PRINT=(CDICT/DICT, TIME)

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário sem registros-C.
TIME	Imprime a hora após cada tabela.

4. **Especificações de subconjunto** (opcional). Essas declarações permitem a seleção de subconjuntos de casos para uma tabela ou conjunto de tabelas.

```
Exemplo:  CLASS    INCLUDE V8=1,2,3,-7,9
```

Há dois tipos de especificações de subconjunto: filtros locais e fatores de repetição. Cada um possui uma função diferente, mas eles são muito similares. Uma especificação pode ser usada como um filtro local para uma ou mais tabelas e como um fator de repetição para outras tabelas.

Regras para codificação

Protótipo: nome declaração

nome
: Nome do subconjunto. 1-8 caracteres alfanuméricos iniciando-se com uma letra. Esse nome deve corresponder exatamente ao nome usado em especificações de análise subseqüentes. Espaços em branco intercalados não serão permitidos. É recomendável que todos os nomes sejam justificados a esquerda.

declaração
: Definição de subconjunto que segue a sintax da declaração de filtro padrão do IDAMS.

Para fatores de repetição, somente uma variável deve ser especificada na expressão.

A maneira como filtros locais e fatores de repetição funcionam é descrita abaixo.

Filtros locais. Uma especificação de subconjunto é identificada com um filtro local para uma tabela ou conjunto de tabelas especificando-se um nome de subconjunto com o parâmetro FILTER. O filtro local opera da mesma maneira que o filtro padrão exceto que ele se aplica somente às especificações de tabela onde são referenciados.

```
Exemplo:  EDUCATN                  INCLUDE V4=0-4,9 AND V5=1
          (nome de subconjunto)    (expressão)
```

No exemplo acima, se EDUCATN é designada como um filtro local na especificação de tabela, a tabela seria produzida incluindo apenas aqueles casos codificados com 0, 1, 2, 3, 4 ou 9 para V4 e 1 para V5.

Fatores de repetição. Uma especificação de subconjunto é identificada como um fator de repetição para uma tabela ou conjunto de tabelas especificando-se o nome do subconjunto com o parâmetro

REPE. Somente uma variável pode ser dada em uma especificação de subconjunto a ser usada como fator de repetição. Fatores de repetição permitem a geração de tabela 3-way onde a variável usada no fator de repetição pode ser considerada como a variável controle ou painel. Usando-se um fator de repetição e um filtro, tabelas 4-way podem ser produzidas.

Expressões INCLUDE fazem com que as tabelas sejam produzidas com a inclusão de cada valor ou intervalo de valores da variável controle usada na expressão. Vírgulas separam os valores ou intervalos. Portanto, se houver n vírgulas na expressão, serão produzidas n+1 tabelas.

```
Exemplo:  EDUCATN                 INCLUDE V4=0-4,9
          (nome do subconjunto)   (expressão)
```

No exemplo acima, se EDUCATN é designado como um fator de repetição, duas tabelas resultarão: uma incluindo casos codificados 0-4 para a variável 4, e outra incluindo casos codificados 9 para a variável 4.

EXCLUDE pode ser usado para produzir tabelas com todos os valores exceto aqueles especificados.

```
Exemplo:  EDUCATN                 EXCLUDE V1=1,4
          (nome de subconjunto)   (expressão)
```

No exemplo acima, se EDUCATN é designado como um fator de repetição, duas tabelas resultarão: uma incluindo todos os valores exceto 1 e o outra incluindo todos os valores exceto 4.

5. **TABLES.** A palavra TABLES nessa linha sinaliza que as especificações vêm a seguir. Deve ser incluído (para separar-se especificações de subconjunto das especificações de tabela) e deve aparecer somente uma vez.

6. **Especificações de tabela.** Especificações de tabela são usadas para descrever as características das tabelas a serem produzidas. As regras de codificação são as mesmas das dos parâmetros. Cada conjunto de especificação de tabelas devem começar em uma nova linha.

```
Exemplos:

R=(V6,1,8) CELLS=FREQS                (Uma tabela univariada).
R=(V6,1,8)  C=(V9,0,4) -              (Uma tabela bivariada com fator de
REPE=SEX  CELLS=(ROWP,FREQS)           repetição, i.e. tabela 3-way).
ROWV=(V5-V9) CELLS=FREQS USTA=MEAN    (Conjunto de tabelas univariadas).
ROWV=(V3,V5) COLV=(V21-V31)  -        (Conjunto de tabelas bivariadas).
  R=(0,1,8) C=(0,1,99)
```

ROWVARS=(lista de variáveis)
: Lista de variáveis de onde tabelas univariadas são requisitadas ou são utilizadas como as linhas em tabelas bivariadas.

COLVARS=(lista de variáveis)
: Lista de variáveis a serem utilizadas como colunas para tabelas bivariadas.

R=(var, rmin, rmax)

var	Número da linha ou variável univariada para uma tabela única. Para fornecer os valores mínimos e máximos para um conjunto de tabelas, faça o número da variável ser zero, e.g. R=(0,1,5); nesse caso, os códigos mínimos e máximos se aplicam a todas as variáveis no parâmetro ROWVARS.
rmin	Código mínimo das variáveis de linha para cálculos estatísticos e percentuais.
rmax	Código máximo das variáveis de linha para cálculos estatísticos e percentuais.

Se o rmin ou rmax é especificado, ambos devem ser especificados. Se somente o número da variável é especificado, valores mínimos e máximos não se aplicam.

C=(var, cmin, cmax)

var Número da variável de coluna para uma tabela bivariada única. Para fornecer os valores máximos e mínimos para um conjunto de tabelas, faça o número da variável ser zero, e.g. C=(o,2,5); nesse caso, os códigos mínimos e máximos se aplicam em todas as variáveis no parâmetro COLVARS.

cmin Código mínimo das variáveis de coluna para cálculos estatísticos e percentuais.

cmax Código máximo das variáveis de coluna para cálculos estatísticos e percentuais.

Se o rmin ou rmax é especificado, ambos devem ser especificados. Se somente o número da variável é especificado, valores mínimos e máximos não se aplicam.

TITLE='título de tabela'

Título a ser impresso no topo de cada tabela nesse conjunto.
Default: Não há título da tabela.

CELLS=(ROWPCT, COLPCT, TOTPCT, **FREQS**/NOFREQS, UNWFREQS, MEAN)

Conteúdos das células das tabelas quando PRINT=TABLES ou WRITE=TABLES é especificado.

ROWP Percentuais para tabelas univariadas ou percentuais baseados em totais de linha para tabelas bivariadas.

COLP Percentuais baseados em totais de colunas para tabelas bivariadas.

TOTP Percentuais baseados no grand total em tabelas bivariadas.

FREQ Contagens de freqüência ponderada (mesmo que o não-ponderado, caso WEIGHT não seja especificado).

UNWF Contagens de freqüência não-ponderada.

MEAN Média da variável especificada por VARCELL.

VARCELL=número de variável

Número de variável para a qual o valor médio deve ser computado para cada célula na tabela.

MDHANDLING=**ALL**/R/C/NONE

Indica quais valores de dados perdidos devem ser excluídos dos cálculos estatísticas e percentuais.

ALL Deleta todos os valores de dados perdidos.

R Deleta todos os valores de dados perdidos para variáveis de linha.

C Deleta todos os valores de dados perdidos para variáveis de coluna.

NONE Não deleta dados perdidos. Nota: casos de dados perdidos são sempre excluídos das estatísticas univariadas.

WEIGHT=número de variável

O número da variável de ponderação se os dados forem ponderados.

FILTER=xxxxxxxx

O nome de 1-8 caracteres da especificação de subconjunto a ser usado como filtro local. Coloque o nome entre aspas simples se ele contiver quaisquer caracteres não-alfanuméricos. Se o nome não faz o match com nenhuma especificação de subconjunto, a tabela será evitada. Letras maiúsculas devem ser usadas para fazer o match do nome da especificação de subconjunto que é automaticamente convertido para letras maiúsculas.

REPE=xxxxxxxx

O nome de 1-8 caracteres da especificação de subconjunto a ser usado como fator de repetição. Coloque o nome entre aspas simples se ele contiver quaisquer caracteres não-alfanuméricos. Se o nome não faz o match com nenhuma especificação de subconjunto, a tabela será evitada. Tabelas serão repetidas para cada grupo de casos especificados. Letras maiúsculas devem ser usadas para fazer o match do nome da especificação de subconjunto que é automaticamente convertido para letras maiúsculas.

USTATS=(MEANSD, MEDMOD)

(Somente tabelas univariadas).

MEAN Imprime média, mínimo, máximo, variância (não-viesada), desvio-padrão, coeficiente de variação, assimetria, curtose, número total de casos ponderados e não-ponderados.

MEDM Imprime mediana e moda (se existirem empates, valores numericamente menores são selecionados).

NTILE=n

(Somente tabelas univariadas).

O n é o número de quantiles a serem calculados; ele deve estar no intervalo 3-10.

STATS=(CHI, CV, CC, LRD, LCD, LSYM, SPMR, GAMMA, TAUA, TAUB, TAUC, EBMSTAT, WILC, MW, FISHER, T)

Se quaisquer estatísticas bivariadas forem impressas ou produzidas, forneça o parâmetro STAT com cada uma das estatísticas desejadas.

Tabelas bivariadas e matriz de saída

CHI Chi-quadrado. (Se MATRIX não é requerida, a seleção de CHI, CV ou CC fará com que os três sejam computados).

CV V de Cramer.

CC Coeficiente de contigência.

LRD Lambda, variável de linha é a variável dependente. (Se MATRIX não é requisitada, a seleção de quaisquer um dos lambdas fará com que os três sejam computados).

LCD Lambda, variável de coluna é a variável dependente.

LSYM Lambda, simétrico.

SPMR Estatística ro de Spearman.

GAMM Estatística Gamma.

TAUA Estatística Tau a. (Se MATRIX não é requisitada, a seleção de quaisquer um dos taus fará com que os três sejam computados).

TAUB Estatística Tau b.

TAUC Estatística Tau c.

Somente tabelas bivariadas

EBMS Estatísticas de Medicina Baseada em Evidência.

WILC Teste de signed rank de Wilcoxon.

MW Teste de Mann-Whitney.

FISH Teste exato de Fisher.

T Testes-t entre todas as combinações de linhas, até um limite de 50 linhas.

DECPCT=**2**/n

Número de decimais, máximo 4, impressos como percentuais.

DECSTATS=**2**/n

Número de decimais impressos para estatísticas: média, mediana, taus, gamma, lambdas e chi-quadrado. Todas as outras estatísticas serão impressas com 2+n decimais (i.e. o default é 4).

WRITE=MATRIX/TABLES

Se um arquivo de saída deve ser gerado, forneça o parâmetro WRITE e o tipo de saída.

MATR Produz as matrizes para estatísticas selecionadas.
Se o parâmetro ROWVARS é especificado, produz uma matriz quadrada para cada estística requerida pelo parâmetro STATS usando todos os pares de variáveis aparecendo na lista.
Se os parâmetros ROWVARS e COLVARS são especificados, produz uma matriz retangular para cada estatística requisitada pelo parâmetro STATS usando cada variável aparecendo na lista ROWVARS emparelhada com cada variável aparecendo na lista COLVARS.

TABL Produz as tabelas de estatísticas requisitadas com o parâmetro CELLS.

PRINT=(**TABLES**/NOTABLES, SEPARATE, ZEROS, CUM, **GRID**/NOGRID,
N, WTDN, MATRIX)

Opções relevantes a tabelas univariadas/bivariadas apenas.

TABL	Imprime tabelas com itens especificados por CELLS.
SEPA	Imprime cada item especificado em CELL como uma tabela separada.
ZERO	Matem as linhas com resultados zeros nas marginais. (Aplicável somente se a tabela possuir mais de 10 colunas e, portanto, deve ser impressa em listas).
CUM	Imprime freqüências marginais de linha e de coluna cumulativas e percentuais. Se os dados são ponderados, figuras são computadas em freqüências ponderadas apenas.
GRID	Imprime o grid ao redor das células de tabelas bivariadas.
NOGR	Suprime o grid ao redor de células de tabelas bivariadas.

Opções relevantes com WRITE=MATRIX apenas.

N	Imprime a matriz de n's para matrizes de estatísticas requeridas.
WTDN	Imprime a matriz de n's ponderados para matrizes de estatísticas requeridas.
MATR	Imprime matrizes de estatísticas especificadas sob STATS.

37.9 Restrições

1. O número máximo de variáveis para freqüências univariadas é 400.

2. A combinação de variáveis e subconjuntos de especificações é sujeito à restrição:

   $5NV + 107NF < 8499$

   onde NF é o número de especificação de subconjunto e NV é o número de variáveis.

3. Valores de código para tabelas univariadas devem estar no intervalo -2.147.483.648 até 2.147.483.647.

4. Valores de códigos de tabelas bivariadas devem estar no intervalo -32.768 até 32.767. Quaisquer valores de código fora desse intervalo são automaticamente recodificados para assumirem os pontos finais do intervalo, e.g. -40.000 se tornará -32.768 e 40.000 se tornará 32.767. Portanto, na especificação de tabela bivariada, 32.767 é o máximo "valor máximo". (Note que uma variável de 5 dígitos com um código de dados perdidos de 99999 terá a linha de dados perdidos de 32.767 nos resultados).

5. A freqüência ponderada ou não-ponderada cumulativa para uma tabela (e para qualquer célula, linha ou coluna) é 2.147.483.647.

6. Máximas dimensões de tabela.

   Bivariadas: 500 códigos de linha, 500 códigos de coluna, 3000 células com entidades não-zero.
   Univariadas: 3000 categorias se freqüências, mediana/moda requisitada; caso contrário, ilimitada.
   Nota: Para uma variável como renda, se houver mais do que 3000 valores únicos de renda, não se pode obter a mediana ou moda sem primeiro transformá-la em variável intervalar.

7. Valores de variável-V não-inteiras em distribuições ou pesos são tratados como se o ponto decimal estivesse ausente, um fator de escala é impresso para cada variável.

8. Testes-t de médias entre linhas são executados somente nas primeiras 50 linhas de uma tabela.

9. Para de matrizes de estatísticas bivariadas de saída, o número máximo de variáveis que podem ser requisitadas para uma linha ou coluna é 95.

10. Se arquivos de saída para tabelas e matrizes são ambos requisitados, eles são produzidos em um mesmo arquivo físico.

11. Não há nenhum jeito de colocar labels em linhas e colunas de tabelas quando variáveis recodificadas são usadas.

37.10 Exemplo

No exemplo abaixo, as seguintes tabelas são requisitadas:

1. Contagem de freqüência para as variáveis V201-V220.
2. Estatísticas univariadas sem tabelas de freqüência para as variáveis V54-V62 e V64. Médias terão uma casa decimal e outras estatísticas, 3 casas decimais.
3. Contagens de freqüências e percentuais com frequências cumulativas e percentuais para variáveis V25-V30 e uma versão agrupada da variável V7. Casos com dados perdidos não devem ser excluídos dos percentuais ou estatísticas. Estatísticas de mediana ou moda requisitadas.
4. Para as categorias de uma única variável V201, contagens de freqüência e a média da variável V54.
5. 8 tabelas bivariadas (com variáveis de linha V25-V28 e variáveis de coluna V29, V30) repetida pelos valores 1 e 2 da variável V10 (sexo), i.e. com sexo como uma variável de painel (controle). Contagens, percentuais de linha, coluna e total estarão em cada célula. Estatísticas chi-quadrado e Taus requisitadas.
6. Tabelas 3-way, usando região (V3) agrupada em 3 categorias com uma variável de painel. Tabelas são restritas aos casos masculino (V10=1). Contagens de freqüências e média da variável V54 aparecem em cada célula.
7. Uma única tabela de contagem de freqüência ponderada, excluindo casos onde ou a variável de linha e/ou variável de coluna assumem valor 9.
8. Matrizes de estatísticas Tau A e Gamma a serem impressas e escritas em um arquivo para todos os pares de variáveis V54-V62. Uma matriz de contagens de casos válidos para cada par de variáveis será impressa.

```
    $RUN TABLES
    $FILES
    PRINT  = TABLES.LST
    FT02   = TREE.MAT                  matrizes de estatísticas
    DICTIN = TREE.DIC                  arquivo Dicionário de entrada
    DATAIN = TREE.DAT                  arquivo Dados de entrada
    $RECODE
        R7=BRAC(V7,0-15=1,16-25=2,26-35=3,36-45=4,46-98=5,99=9)
        NAME R7'GROUPED V7'
    $SETUP
    TABLE EXAMPLES
    BADDATA=MD1
    MALE     INCLUDE V10=1
    SEX      INCLUDE V10=1,2
    REGION   INCLUDE V3=1-2,3-4,5
    MD       EXCLUDE V19=9 OR V52=9
    TABLES
1.  ROWV=(V201-V220)  TITLE='Frequency counts'
2.  ROWV=(V54-V62,V64)  USTATS=MEANSD  PRINT=NOTABLES  DECSTAT=1
3.  ROWV=(V25-V30,R7)   USTATS=MEDMOD  CELLS=(FREQS,UNWFREQS,ROWP)  -
        WEIGHT=V9  PRINT=CUM  MDHAND=NONE
4.  R=(V201,1,3)  CELLS=(FREQS,MEAN)  VARCELL=V54
5.  ROWV=(V25-V28)  COLV=(V29-V30)  -
        CELLS=(FREQS,ROWP,COLP,TOTP)  STATS=(CHI,TAUA)  REPE=SEX
6.  ROWV=(V201-V203)  COLV=V206  -
        CELLS=(FREQS,MEAN)  VARCELL=V54  REPE=REGION  FILT=MALE
7.  R=V19  C=V52  WEIGHT=V9  FILT=MD
8.  ROWV=(V54-V62)  STATS=(TAUA,GAMMA)  PRINT=(MATRIX,N)  WRITE=MATRIX
```

Capítulo 38

Tipologia e Classificação Ascendente (TYPOL)

38.1 Descrição Geral

TYPOL cria uma variável de classificação resumindo um grande número de variáveis. O uso de uma variável inicial de classificação, definida "a priori" (variável chave), ou uma amostra aleatória de casos, ou uma amostra por passos são permitidas serem o núcleo inicial dos grupos. Um procedimento iterativo refina os resultados ao estabilizar os núcleos. Os grupos finais constituem as categorias da variável de classificação procuradas. O número de grupos da tipologia pode ser reduzido usando-se um algoritmo de classificação ascendente hierárquica.

As **variáveis ativas** são as variáveis em cuja base o agrupamento e reagrupamento de casos é executado. Pode-se também buscar pelas estatísticas principais de outras variáveis dentro dos grupos construídos de acordo com as variáveis ativas. Tais variáveis (que não possuem influência na construção dos agrupamentos) são chamadas de **variáveis passivas**.

TYPOL aceita tanto variáveis quantitativas quanto qualitativas, as últimas sendo tratadas como quantitativas depois de uma completa dicotomização de suas categorias respectivas, resultando na construção de tantas variáveis dicotomizadas (1/0) quanto o número de categorias da variável qualitativa. É também possível padronizar as variáveis ativas (as variáveis quantitativas, e a qualitativa depois da dicotomização).

TYPOL opera em dois passos:

1. **Construção de uma tipologia inicial.** O program constrói uma tipologia de n grupos, como requisitado pelo usuário, dos casos caracterizados por um número de variáveis (consideradas quantitativas). O usuário pode selecionar a maneira que uma configuração inicial é estabelecida (ver o parâmetro INICIAL), e também o tipo de distância (ver o parâmetro DTYPE) usado pelo programa para calcular a distância entre casos e grupos.

2. **Classificação ascendente posterior** (opcional). Se o usuário desejar uma tipologia em um menor número de grupos, o programa, usando um algoritmo de classificação ascendente hierárquica, reduz, um a um, o número de grupos até o número especificado pelo usuário.

38.2 Características Padrão do IDAMS

Seleção de casos e variáveis. O filtro padrão está disponível para selecionar um subconjunto de casos dos dados de entrada. As variáveis são especificadas com parâmetros.

Transformando dados. Declarações de Recode podem ser usadas.

Ponderando dados. Uma variável pode ser usada para ponderar os dados de entrada; essa variável de ponderação pode conter valores inteiros ou decimais. Quando o valor da variável de ponderação para um

caso for zero, negativo, perdido ou não-numérico, então o caso é evitado; o número de casos tratados dessa maneira é impresso.

Tratamento de dados perdidos. O parâmetro MDVALUES está disponível para indicar quais valores de dados perdidos, se houver, devem ser usados para checar a existência de dados perdidos. Casos com dados perdidos nas variáveis quantitativas podem ser excluídos da análise (ver o parâmetro MDHANDLING).

38.3 Resultados

Dicionário de entrada. (Opcional: ver o parâmetro PRINT). Registros de descrição de variáveis, e registros-C, se houver, somente para variáveis usadas na execução.

Tipologia inicial

Construção de uma tipologia inicial. (Opcional: ver o parâmetro PRINT).

O reagrupamento de grupos iniciais, seguido por uma tabela de números de referências-cruzadas atribuídos aos grupos antes e depois da construção dos grupos iniciais.

Tabelas mostrando a redistribuição de casos entre uma iteração e a seguinte, e dando o percentual do número total de casos agrupados apropriadamente.

Evolução do percentual da variância explicada de uma iteração a outra.

Características de distâncias por grupos. O número de casos em cada grupo inicial da tipologia, juntamente com o valor da média e o desvio-padrão das distâncias.

Classificação de distâncias. (Opcional: ver o parâmetro PRINT). Tabela mostrando, dentro de cada grupo, a distribuição de casos através de quinze intervalos contínuos, esses intervalos são:

diferentes para cada grupo (primeira tabela),

idênticos para todos os grupos (segunda tabela).

Características globais de distâncias. O número total de casos, com a média e desvio-padrão totais das distâncias.

Estatísticas de resumo. A média, desvio-padrão e o peso da variável para as variáveis quantitativas e para categorias de variáveis qualitativas ativas.

Descrição de tipologia resultante. Para cada grupo de tipologia, seu número e o percentual de casos pertencendo a ele são impressos primeiro. Então, as estatísticas são fornecidas, variável por variável, na seguinte ordem: (1) variáveis ativas quantitativas; (2) variáveis passivas quantitativas; (3) variáveis ativas qualitativas; (4) variáveis passivas qualitativas.

Para cada variável quantitativa, são dadas a quantidade de variância explicada, seu valor médio total e, dentro de cada grupo da tipologia, seu valor médio e desvio-padrão.

Para cada categoria da variável qualitativa, são dadas inicialmente sua quantidade de variância explicada e o percentual de casos pertencendo a ela; então, dentro de cada grupo de tipologia são impressos: verticalmente, o percentual de casos ao longo das categorias da variável na primeira linha e horizontalmente, o percentual de casos ao longo dos grupos de tipologia (percentuais de linha) na segunda linha (opcional: ver o parâmetro PRINT).

Resumo da quantidade de variância explicada pela tipologia. Os seguintes percentuais da variância explicada são dados:

a variância explicada pelas variáveis mais discriminantes, i.e. aquelas que, tomadas juntas, são responsáveis por oitenta por cento da variância explicada,

a quantidade média da variância explicada pelas variáveis ativas,

a quantidade média da variância explicada por todas as variáveis juntas,

a quantidade média da variância explicada pelas variáveis mais discriminantes juntamente com a proporção dessas variáveis.

Nota: Quando as variáveis qualitativas aparecem em tabelas, os primeiros 12 caracteres do nome da variável são impressos junto com o valor de código identificando a categoria. Quando as variáveis quantitativas aparecem em tabelas, todos os 24 caracteres do nome da variável são impressos.

Classificação hierárquica ascendente

Tabela de raízes quadradas de deslocamentos e distâncias calculadas para cada par de grupos. (Opcional: ver o parâmetro PRINT).

Tabela de reagrupamento No. 1. Estatísticas de resumo para as variáveis ativas quantitativas e categorias de variáveis ativas qualitativas para grupos envolvidos no reagrupamento.

Descrição da nova tipologia resultante. (Opcional: ver o parâmetro LEVELS). As mesma informação acima.

Resumo da quantidade de variância explicada pela nova tipologia. A mesma informação acima. Note aqui que a quantidade média de variância explicada pelas variáveis mais discriminante antes do reagrupamento.

O sumário da classificação hierárquica ascendente é impresso depois de cada reagrupamento até um número de grupos especificados pelo usuário.

Três diagramas mostrando o percentual de variância explicada como uma função do número de grupos das tipologias sucessivas, por vez, para:
todas as variáveis,
as variáveis ativas,
as variáveis explicando 80% da variância antes dos reagrupamentos acontecerem.

Perfis de cada grupo da tipologia. (Opcional: ver o parâmetro PRINT). Esses perfis são impressos e plotados para todos os grupos da primeira tipologia resultante e, então, para os grupos obtidos a cada reagrupamento.

Árvore hierárquica é produzida no final.

38.4 Dataset de Saída

Um dataset de "variável de classificação" para a primeira tipologia resultante pode ser requisitado e, então, produzido na forma de um arquivo de dados descrito por um dicionário do IDAMS (ver o parâmetro WRITE e o capítulo "Dados em IDAMS"). Ele contém a variável de ID de caso, as variáveis transferidas, a variável de classificação ("GROUP NUMBER") e, para cada caso, sua distância multiplicada por 1000 de cada categoria da variável de classificação, chamada "n GROUP DISTANCE". As variáveis são numeradas começando de um e incrementando por um na seguinte ordem: variável de ID de caso, variáveis transferidas, variável de classificação e variáveis de distância.

38.5 Matriz de Configuração de Saída

Uma matriz de configuração de saída pode ser opcionalmente escrita na forma de uma matriz retangular do IDAMS (ver parâmetro WRITE). Ver o capítulo "Dados em IDAMS" para uma descrição do formato. Essa matriz fornece, linha por linha, para cada variável quantitativa e para cada categoria de variáveis ativas qualitativas, seus valores médios ao longo dos grupos e desvio-padrão geral para a tipologia inicial, i.e. antes do reagrupamento acontecer. Os elementos da matriz são escritos em formato 8F9.3. Registros de dicionário são escritos.

38.6 Dataset de Entrada

A entrada é um arquivo Dados descrito por um dicionário do IDAMS. Todas as variáveis de análise devem ser numéricas; elas podem possuir valores inteiros ou decimais. A variável de ID de caso e variáveis a serem transferidas podem ser alfabéticas.

38.7 Matriz de Configuração de Entrada

A matriz de configuração de entrada deve estar na forma de uma matriz retangular do IDAMS. Ver o capítulo "Dados em IDAMS" para uma descrição do formato. Essa matriz é opcional e fornece uma configuração inicial para ser usada nas computações. As estatísticas incluídas devem ser valores médios para as variáveis quantitativas e proporções (não percentagens) para as categorias das variáveis qualitativas (e.g. .180 ao invés de 18.0 por cento). Uma matriz de configuração produzida pelo programa em uma execução prévia pode servir de configuração de entrada.

38.8 Estrutura de Setup

```
$RUN TYPOL

$FILES
     Especificações de arquivo

$RECODE (optional)
     Declarações de Recode

$SETUP
     1. Filtro (opcional)
     2. Título
     3. Parâmetros

$DICT (condicional)
     Dicionário

$DATA (condicional)
     Dados

$MATRIX (condicional)
     Matriz de configuração de entrada


Arquivos:
FT02       matriz de configuração de saída se WRITE=CONF for especificado
FT09       matriz de configuração de entrada se INIT=INCONF for especificado
           (omitir se $MATRIX é usado)
DICTxxxx   dicionário de entrada (omitir se $DICT é usado)
DATAxxxx   dados de entrada (omitir se $DATA é usado)
DICTyyyy   dicionário de saída se WRITE=DATA é especificado
DATAyyyy   dados de saída se WRITE=DATA é especificado
PRINT      resultados (default  IDAMS.LST)
```

38.9 Declarações de Controle de Programa

Referir-se ao capítulo "O Arquivo Setup do IDAMS" para descrições mais detalhadas das declarações de controle de programa, itens 1-3 abaixo.

1. **Filtro** (opcional). Seleciona um subconjunto de casos a ser usado na execução.

   ```
   Exemplo: INCLUDE V1=10-40,50
   ```

2. **Título** (mandatório). Uma linha contendo até 80 caracteres para dar nome aos resultados.

```
Exemplo: FIRST CONSTRUCTION OF CLASSIFICATION VARIABLE
```

3. **Parâmetros** (mandatório). Para selecionar opções de programa.

```
Exemplo: MDHAND=ALL AQNTV=(V12-V18) DTYP=EUCL  -
            PRINT=(GRAP,ROWP,DIST) INIG=5 FING=3
```

INFILE=**IN**/xxxx
Um sufixo ddanme de 1-4 caracteres para os arquivos Dicionário e Dados de entrada. Default ddnames: DICTIN, DATAIN.

BADDATA=**STOP**/SKIP/MD1/MD2
Tratamento de valores de dados não-numéricos. Ver o capítulo "O Arquivo Setup do IDAMS".

MAXCASES=n
O número máximo de casos (depois da filtragem) a serem usados do arquivo de entrada. Default: Todos os casos serão usados.

AQNTVARS=(lista de variáveis)
Uma lista de variáveis especificando variáveis ativas quantitativas.

PQNTVARS=(lista de variáveis)
Uma lista de variáveis especificando variáveis passivas quantitativas.

AQLTVARS=(lista de variáveis)
Uma lista de variáveis especificando variáveis ativas qualitativas.

PQLTVARS=(variable list)
Uma lista de variáveis especificando variáveis passivas qualitativas.

MDVALUES=**BOTH**/MD1/MD2/NONE
Quais valores de dados perdidos devem ser utilizados para as variáveis acessadas nessa execução. Ver o capítulo "O Arquivo Setup do IDAMS".

MDHANDLING=**ALL**/QUALITATIVE/QUANTITATIVE

ALL	Casos com valores de dados perdidos em variáveis quantitativas serão evitados e códigos de dados perdidos em variáveis qualitativas serão excluídos da análise.
QUAL	Valores de dados perdidos em variáveis qualitativas serão excluídos da análise.
QUAN	Casos com valores de dados perdidos em variáveis quantitativas serão evitados.

REDUCE
Padronização das variáveis ativas, tanto quantitativas quanto qualitativas.

WEIGHT=número de variável
O número da variável de ponderação, se os dados forem ponderados.

DTYPE=**CITY**/EUCLIDEAN/CHI

CITY	Distância city-block.
EUCL	Distância euclidiana.
CHI	Distância chi-quadrado.

Nota: A respeito da escolha do tipo de distância, é recomendável usar:

- A distância city-block quando algumas variáveis ativas são qualitativas e outras quantitativas,

- A distância euclidiana quando as variáveis ativas são todas quantitativas (com padronização se elas não estiverem medidas na mesma escala),
- A distância chi-quadrado quando as variáveis ativas são todas qualitativas.

INIGROUP=n

Número de grupos iniciais. Se uma variável chave deve servir como base para a tipologia, e se o número de grupos iniciais especificados aqui é maior do que o valor máximo da variável chave, o programa automaticamente corrige isso. Também, se há certas categorias com casos de zeros, o número de grupos iniciais será o número de categorias não vazias.
Não há default.

FINGROUP=**1**/n

Número de grupos finais.

INITIAL=**STEPWISE**/RANDOM/KEY/INCONF

A maneira em que a configuração inicial é estabelecida.

STEP	Amostra por passos.
RAND	Amostra aleatória.
KEY	Perfil de grupos iniciais é criado de acordo com variáveis chave.
INCO	Um perfil "a priori" de grupos iniciais é dado em um arquivo de configuração de entrada. Nota: Variáveis incluídas na configuração de entrada devem corresponder exatamente as variáveis fornecidas com os parâmetros AQNTV e/ou AQLTV.

STEP=**5**/n

Se amostra por passos de casos é requisitada (INIT=STEP), n é o comprimento do passo.

NCASES=n

Se a amostra aleatória de casos é requisitada (INIT=RAND), n é o número de casos (não-ponderados) no arquivo de entrada, ou uma boa estimativa disso.
Não há default; deve ser especificado se INIT=RAND.

KEY=número de variável

Se uma variável chave é usada para construir grupos iniciais (INIT=KEY), esse é o número da variável chave.
Não há default; deve ser especificado se INIT=KEY.

ITERATIONS=**5**/n

Número máximo de iterações para convergência do perfil de grupo.

REGROUP=**DISPLACEMENT**/DISTANCE

DISP	Reagrupamento é baseado em mínimo deslocamento.
DIST	Reagrupamento é baseado em distância mínima.

WRITE=(DATA, CONFIG)

DATA	Cria um dataset do IDAMS contendo a variável de ID de caso, variáveis transferidas, variável de classificação e variáveis de distância.
CONF	Produz a matriz de configuração em um arquivo.

OUTFILE=**OUT**/yyyy

Um sufixo ddname de 1-4 caracteres para os arquivos Dicionário e Dados de saída.
Default ddnames: DICTOUT, DATAOUT.

IDVAR=número de variável

Variável a ser transferida para o dataset de saída para identificar casos.
Obrigatório se WRITE=DATA é especificado.

TRANSVARS=(lista de variáveis)
Variáveis adicionais (até 99) a serem transferidas para o dataset de saída.

LEVELS=(n1, n2, ...)
Imprime descrição da tipologia resultante para o número de grupos especificados.
Default: Descrição é impressa depois de cada reagrupamento.

PRINT=(CDICT/DICT, OUTCDICT/OUTDICT, INITIAL, TABLES, GRAPHIC, ROWPCT, DISTANCES)

CDIC	Imprime o dicionário de entrada para as variáveis acessadas com registros-C, se houver.
DICT	Imprime o dicionário de entrada sem registros-C.
OUTC	Imprime o dicionário de saída com registros-C, se houver.
OUTD	Imprime o dicionário de saída sem registros-C.
INIT	Imprime a história da construção da tipologia inicial.
TABL	Imprime duas tabelas com classificação de distâncias.
GRAP	Imprime o gráfico dos perfis.
ROWP	Imprime percentuais de linha para categorias de variáveis qualitativas.
DIST	Imprime tabela de distâncias e deslocamentos para cada reagrupamento.

38.10 Restrições

1. Número máximo de grupos iniciais é 30.
2. Número total máximo de variáveis é 500, incluindo variável de ponderação, variável chave, variáveis a serem transferidas, variáveis de análise (variáveis quantitativas + número de categorias para variáveis qualitativas) e variáveis usadas temporariamente em declarações de Recode.
3. Se a variável de ID ou a variável a ser transferida é alfabética com largura > 4, somente os quatro primeiros caracteres são usados.
4. Variáveis-R não podem ser usadas como variáveis ID ou como variáveis a serem transferidas.

38.11 Exemplos

Exemplo 1. Criação de uma variável de classificação resumindo 5 variáveis quantitativas e 4 qualitativas usando a distância city-block; configuração inicial será estabelecida por uma seleção aleatória de casos; classificação se inicia com 6 grupos e terminará com 3 grupos; reagrupamento será baseado em distância mínima; dados perdidos serão excluídos da análise.

```
$RUN TYPOL
$FILES
PRINT  = TYPOL1.LST
DICTIN = A.DIC                          arquivo Dicionário de entrada
DATAIN = A.DAT                          arquivo Dados de entrada
$SETUP
SEARCHING FOR NUMBER OF CATEGORIES IN A CLASSIFICATION VARIABLE
AQNTV=(V114,V116,V118,V120,V122)  AQLTV=(V5-V7,V36)  REDU  -
   INIG=6  FING=3  INIT=RAND  NCAS=1200  -
   REGR=DIST  PRINT=(GRAP,ROWP,DIST)
```

Exemplo 2. Gerando uma variável de classificação do Exemplo 1 com 4 categorias; a variável deve ser escrita em um arquivo; variáveis V18 e V34 são usadas como passivas quantitativas e variáveis V12 e V14 como passivas qualitativas.

```
$RUN TYPOL
$FILES
PRINT   = TYPOL2.LST
DICTIN  = A.DIC                          arquivo Dicionário de entrada
DATAIN  = A.DAT                          arquivo Dados de entrada
DICTOUT = CLAS.DIC                       arquivo Dicionário de saída
DATAOUT = CLAS.DAT                       arquivo Dados de saída
$SETUP
GENERATING A CLASSIFICATION VARIABLE
AQNTV=(V114,V116,V118,V120,V122)  AQLTV=(V5-V7,V36)  REDU  -
   PQNTV=(V18,V34)  PQLTV=(V12,V14)  -
   INIG=6  FING=4  INIT=RAND  NCAS=1200 -
   REGR=DIST  PRINT=(GRAP,ROWP) WRITE=DATA  IDVAR=V1
```

Parte V

Análise Interativa de Dados

Capítulo 39

Tabelas Multidimensionais e suas Apresentações Gráficas

39.1 Visão Geral

O componente interativo "Tabelas multidimensionais" do WinIDAMS permite que você visualize e especifique tabelas multidimensionais com freqüências, percentuais de linha, coluna e total, estatísticas univariadas (soma, contagem, média, máximo, mínimo, variância, desvio-padrão) de variáveis adicionais, e estatísticas bivariadas. Variáveis em linhas e/ou colunas podem ser aninhadas (máximo de 7 variáveis) ou elas podem ser colocadas no mesmo nível. Construção de uma tabela pode ser repetida para cada valor de até três variáveis de "página". Cada página da tabela pode também ser impressa, ou exportada em formato livre (delimitada por vírgula ou caracter de tabulação) ou em formato HTML.

Datasets do IDAMS usados como entrada devem ter o mesmo nome dos arquivos Dicionário e Dados com extensões .dic e .dat respectivamente.

Apenas um dataset pode ser usado por vez, i.e. abrindo-se um outro dataset, automaticamente fecha o outro que estava sendo usado.

39.2 Preparação da Análise

Seleção dos dados. Um dataset selecionado para a construção de tabelas multidimensionais está disponível até que seja modificado ao se ativar, novamente, o componente "Tabelas multidimensionais". A caixa de diálogo permite que você escolha um arquivo Dados de uma lista de arquivos Dados recentemente usados (Recente) ou de um folder qualquer (Existente). O folder Dados da aplicação corrente é o default. Selecionando em "Files of type:" a opção "Arquivos Dados (*.dat)" mostra somente arquivos Dados do IDAMS.

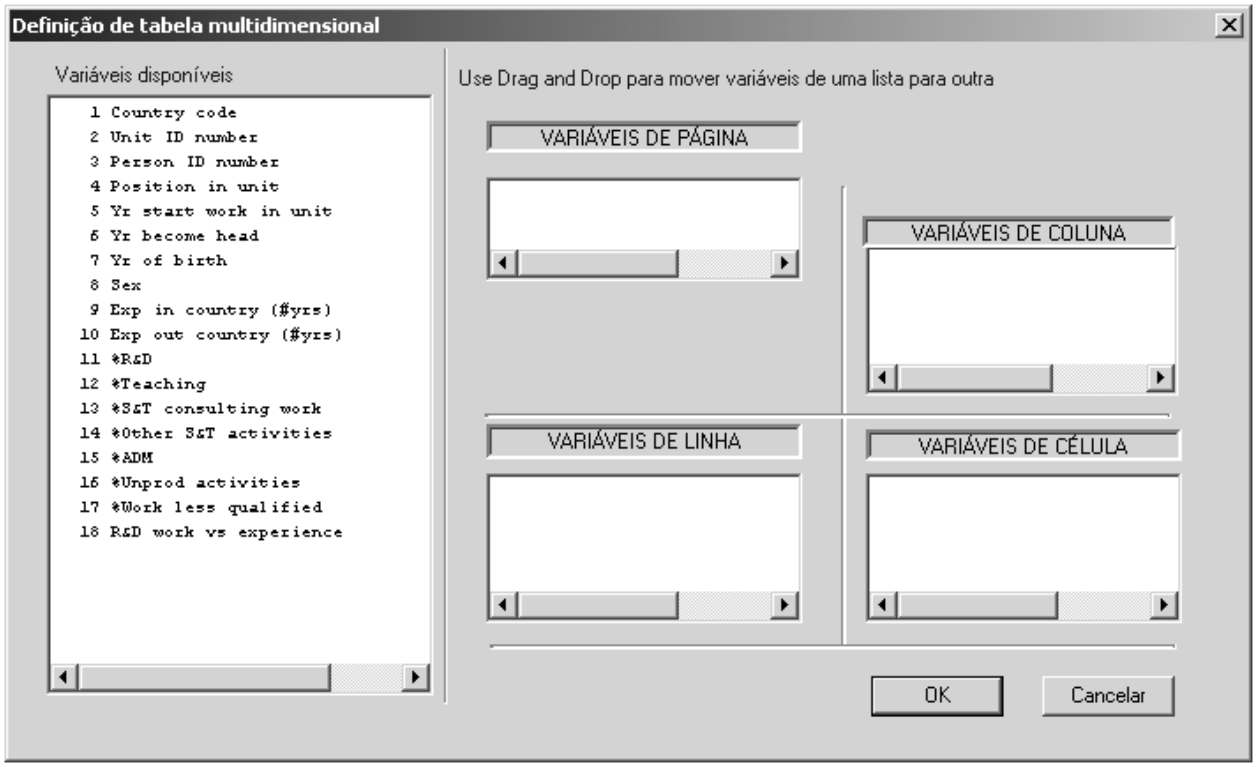
Seleção de variáveis. A seleção de um conjunto de dados para análise invoca uma caixa de diálogo para a definição da tabela. Você será apresentado a uma lista de variáveis disponíveis e quatro janelas para especificar variáveis para diferentes finalidades. Use a técnica "Drag and Drop" para movimentar variáveis entre e/ou dentro das janelas requisitadas.

Variáveis de página são utilizadas para construir páginas separadas da tabela para cada valor distinto de cada variável na vez, e para todos os casos tomados juntos (Página total). Casos inclusos em uma página particular possuem todos os mesmos valores na variável de página. Variáveis de página não são nunca aninhadas. A ordem em que as variáveis são especificadas determina a ordem em que as páginas são colocadas na janela Tabela.

Variáveis de linha são as variáveis cujos valores são usados para definir linhas de tabela. A ordem delas determina a seqüência do uso do aninhamento.

Variáveis de coluna são as variáveis cujos valores são usados para definir as colunas. A ordem delas determina a seqüência do uso do aninhamento.

Variáveis de célula são variáveis cujos valores são usados para calcular estatísticas univariadas (e.g. média) nas células da tabela. A ordem que elas são especificadas determina a ordem das suas aparições na tabela. Devem haver até 10 variáveis de célula.



Aninhamento. Se mais de uma variável de linha e/ou coluna são especificadas, por default, elas são aninhadas. Para usá-las seqüencialmente, no mesmo nível, dê um clique duplo na variável na lista da variável de linha ou de coluna e marque a opção para tratamento a um mesmo nível. Note: Essa opção não está disponível para a primeira variável em uma lista.

Percentagens. Percentagens em cada célula (linha, coluna ou total) podem ser obtidas ao dar-se um clique duplo na última variável de linha aninhada na janela de definição de tabela e selecionar o tipo de percentagens requeridas.

Estatísticas univariadas. Diferentes estatísticas (soma, contagem, média, máximo, mínimo, variância, desvio-padrão) para cada uma das variáveis de célula podem ser obtidas ao dar-se um clique duplo na variável na janela de definição de tabela e marcar-se as estatísticas requeridas. Fórmulas para cálculo da média, variância e desvio-padrão podem ser encontradas na seção "Estatísticas Univariadas" do capítulo "Tabelas Univariadas e Bivariadas". No entanto, elas precisam ser ajustadas, pois os casos não são ponderados.

Tratamento de dados perdidos. O tratamento de dados perdidos default é aplicado na primeira construção da tabela. Então, ele pode ser mudado utilizando-se o menu Mudar.

A opção ***Valores de dados perdidos*** é usada para indicar quais valores de dados pedidos, se houverem, devem ser usados para checar a existência de dados perdidos em variáveis de linha e de coluna.

Ambos	Valores serão checados em relação aos códigos de MD1 e em relação ao intervalo de códigos definidos por MD2.
MD1	Valores serão checados somente em relação aos códigos de MD1.
MD2	Valores de variáveis serão checados somente em relação ao intervalo de códigos definidos por MD2.
Neuhum	Códigos MD não serão utilizados. Todos os valores de dados serão considerados válidos.

Por default. ambos os códigos MD são usados.

A opção ***Manuseio de dados perdidos*** é usada para indicar quais valores de dados perdidos devem ser excluídos do cômputo dos percentuais e das estatísticas bivariadas.

Todos	Delete todos os valores de dados perdidos.
Linha	Delete os valores de dados perdidos para variáveis de linha.
Coluna	Delete os valores de dados perdidos para variáveis de coluna.
Neuhun	Não delete valores de dados perdidos.

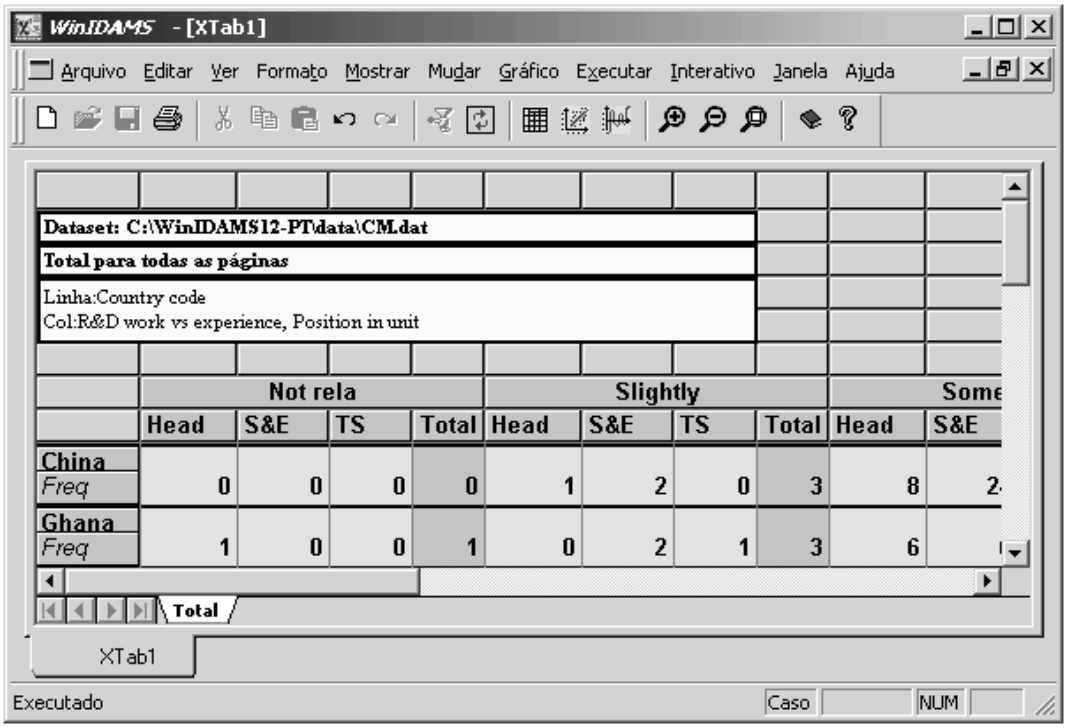
Por default, todos os valores de dados perdidos são deletados.

Nota: Casos com dados perdidos em variáveis de célula são sempre excluídos do cálculo das estatísticas univariadas. A exclusão é feita célula a célula, separadamente para cada variável de célula. Portanto, o número de casos válidos pode não ser igual à freqüência da célula. A estatística "Contagem" mostra o número de casos válidos.

Mudando a definição de tabela. O comando de menu Mudar/Especificação chama a caixa de diálogo com a definição de tabela ativa. Você pode mudar variáveis para análise, os seus aninhamentos, como também as percentagens e estatísticas univariadas requisitadas. Clicando em OK substitui a tabela ativa por uma nova.

39.3 Janela Tabelas Multidimensionais

Depois da seleção de variáveis e de um clique em OK, a janela Tabelas multidimensionais aparece na janela de documento do WinIDAMS. Por default, freqüências e valores médios para todas as células são expostas. Se variáveis de página são especificadas, labels de código (ou códigos) dessas variáveis são mostradas em tabs no final da tabela. Uma página em particular pode ser acessada através de um clique na label requisitada (código).



Modificando a aparência da página. A aparência de cada página pode ser modificada separadamente, com as mudanças aplicando-se exclusivamente à pagina ativa.

As seguintes modificações são possíveis:

- *Aumentando o tamanho da fonte* - use o comando de menu Ver/Zoom In ou o botão Zoom In da barra de ferramentas.
- *Diminuindo o tamanho da fonte* - use o comando de menu Ver/Zoom Out ou o botão Zoom Out da

barra de ferramentas.

- *Redefinindo o tamanho de fonte default* - use o comando de menu Ver/100% ou o botão 100% da barra de ferramentas.

- *Aumentando/Diminuindo a largura de uma coluna* - coloque o cursor do mouse na linha que separa as duas colunas no início da coluna até que ele se torne uma barra vertical com duas flechas e movimente-a para a direita/esquerda mantendo clicado o botão esquerdo do mouse.

- *Minimizando a largura de colunas* - marque as colunas requeridas e use o comando de menu Formato/Mudar tamanho de colunas.

- *Aumentando/Diminuindo a altura das linhas* - coloque o cursor do mouse na linha que separa as duas linha no início da linha até que ele se torne uma barra horizontal com duas flechas e movimente-a para cima/baixo mantendo clicado o botão esquerdo do mouse.

- *Minimizando a altura de linhas* - marque as linhas requeridas e use o comando de menu Formato/Mudar tamanho de linhas.

- *Ocultando colunas/linhas* - diminua a largura/altura de uma coluna/linha até zero. Para mostrar uma coluna/linha ocultada, coloque o mouse na linha onde ela está oculta no início da coluna/linha até que ele se torne uma barra vertical com duas flechas e, então, dê um clique duplo o botão esquerdo do mouse.

Além disso, o comando Formato/Estilo dá acesso a um número de possibilidades de **formatação de tabela** como: seleção de fontes, tamanho de fontes, cores, etc. para célula ativa ou para todas as células em uma linha ativa.

Estatísticas bivariadas. Estatísticas bivariadas (chi-quadrado, coeficiente Phi, coeficiente de contingência, V de Cramer, Taus, Gamma, Lambdas e D de Sormer) são computadas para cada tabela (cada página). Use o comando de menu Mostrar/Estatísticas para mostrá-las no final da tabela. Se necessário, essa operação deve ser repetida para cada página separadamente. Fórmulas para calcular estatísticas bivariadas podem ser encontradas na seção "Estatísticas Bivariadas" do capítulo "Tabelas Univariadas e Bivariadas".
Note que estatísticas são calculadas somente quando há uma variável de linha e uma de coluna.

Imprimindo uma página de tabela. O conteúdo completo da página ativa ou partes desejadas somente podem ser impressos usando o comando Arquivo/Imprimir. Se você quiser imprimir apenas algunas colunas e/ou linhas, oculte as outras colunas/linhas primeiro. As colunas/linhas mostradas serão impresas.

Exportando uma página de tabela. O conteúdo completo da página ativa ou partes desejadas somente podem ser exportados em formato livre (delimitado por vírgula ou caracter de tabulação) ou em formato HTML. Use o comando Arquivo/Exportar e selecione o formato requerido. Se você quiser exportar apenas algunas colunas e/ou linhas, oculte as outras colunas/linhas primeiro. As colunas/linhas mostradas serão exportar.

39.4 Apresentação Gráfica de Tabelas Univariadas/Bivariadas

Freqüências exibidas em uma página de tabelas univariadas/bivariadas podem ser apresentadas graficamente usando um dos 24 estilos a sua disposição. A construção do gráfico é iniciada pelo comando de menu Gráfico/Criar. Esse comando chama a caixa de diálogo para selecionar o estilo do gráfico para a página ativa. Além disso, você pode requisitar uma transformação logarítmica de freqüências, e pode fornecer uma legenda para cores e símbolos usados no gráfico.

Gráficos projetados não podem ser manipulados. Contudo, eles podem ser salvos em um dos dois formatos: JPEG formato de troca de arquivo (.jpg) ou formato de Windows Bitmap (.bmp), usando os comandos relevantes no menu Arquivo. Eles podem também ser copiados para o Clipboard (o comando Editar/Copiar, botão de barra de ferramentas ou teclas de atalho Ctrl/C) e passados para qualquer editor de texto.

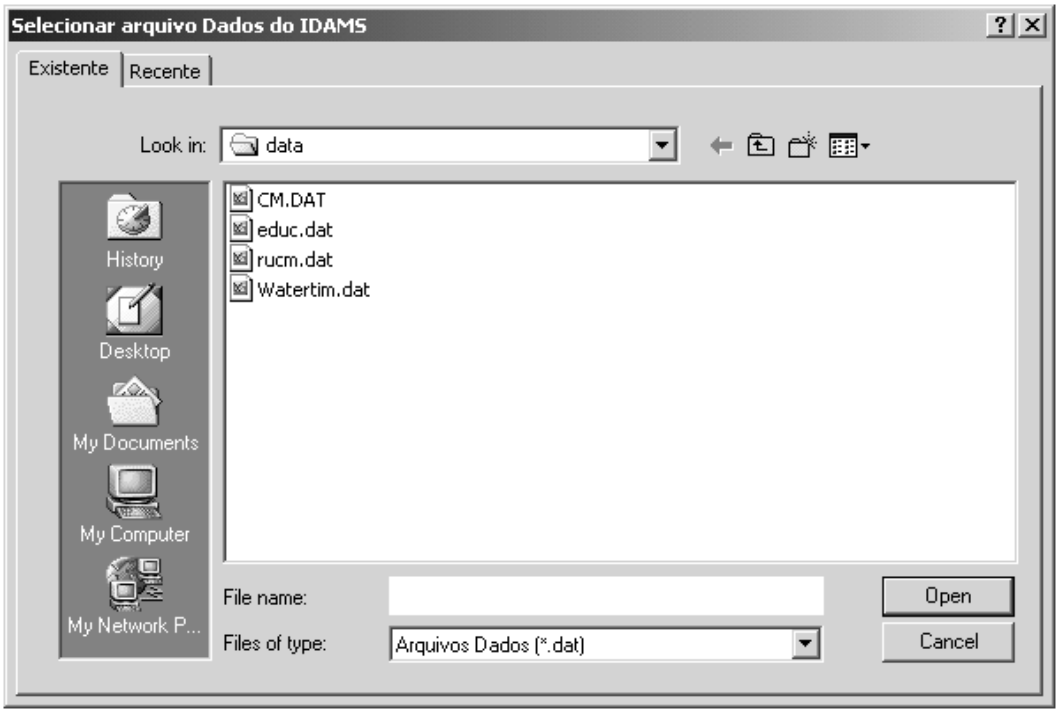
Deve ser notado aqui novamente que somente freqüências de linhas e colunas exibidas, i.e. não de linhas e/ou colunas que foram ocultas, são usadas para essa apresentação.

39.5 Como Fazer uma Tabela Multidimensional

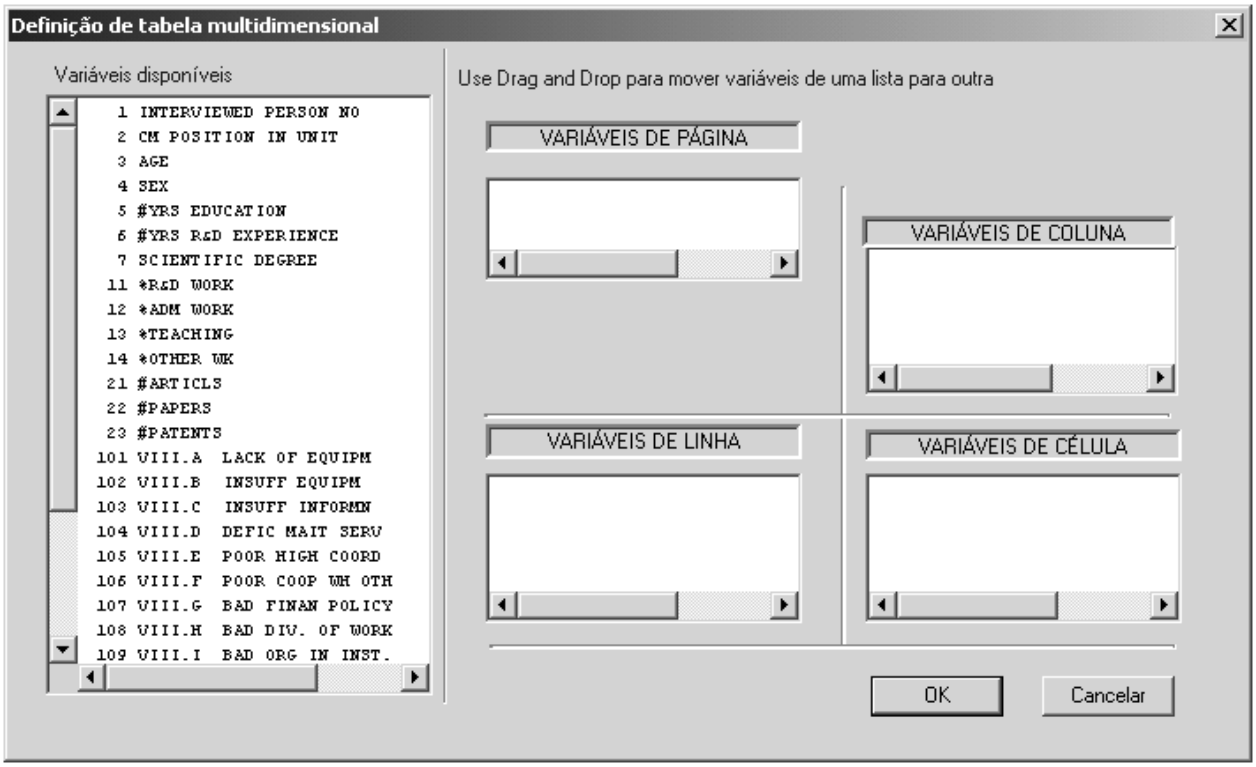
Nós vamos usar o dataset "rucm" ("rucm.dic" é o arquivo Dicionário e "rucm.dat" é o arquivo Dados) que está no folder Dados default e que está instalado no WinIDAMS.

Nós vamos construir uma tabela three-way com duas variáveis de linha aninhadas ("SCIENTIFIC DEGREE" e "SEX") e uma variável de célula ("AGE") para o qual vamos requisitar a média, máximo e mínimo.

- Clique em Interactivo/Tabelas multidimensionais. Esse comando abre um diálogo para selecionar um arquivo Dados do IDAMS.



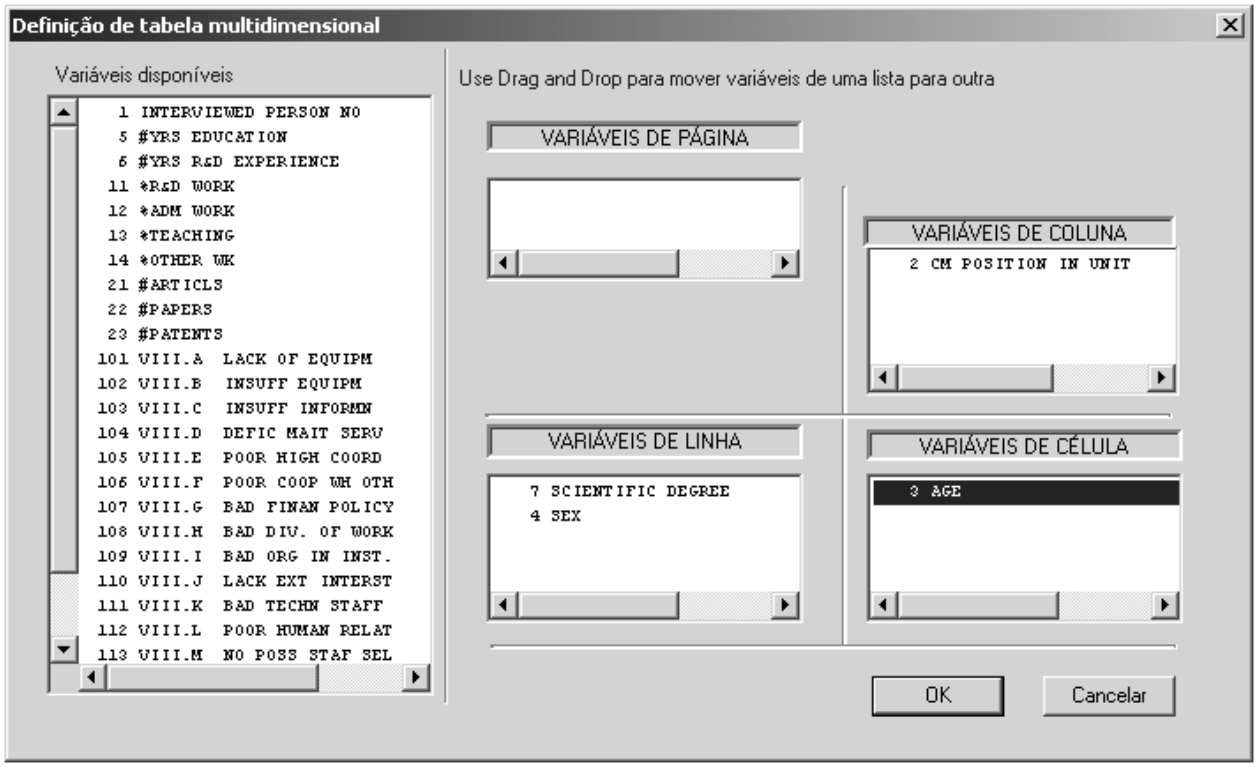
- Clique em rucm.dic e Abrir. Você vê agora um diálogo para especificar as variáveis que você deseja usar na tabela multidimensional.



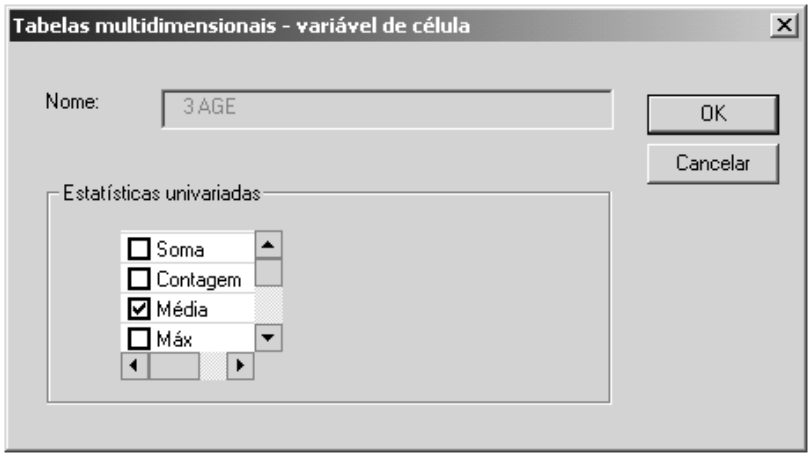
- Selecione as variáveis “SCIENTIFIC DEGREE” e “SEX” como VARIÁVEL DE LINHA, “CM POSITION IN UNIT” como VARIÁVEL DE COLUNA e “AGE” como VARIÁVEL DE CÉLULA.

  Use a técnica “Drag and Drop” do mouse para movimentar as variáveis (pressione o botão esquerdo do mouse na variável que você deseja mover, mantenha o botão pressionado enquanto você move a variável solte na lista de variável para onde você quer mover a variável). Muitas variáveis podem ser selecionadas e movidas simultaneamente de uma lista para a outra (matenha apertada a tecla Ctrl quando selecionar).

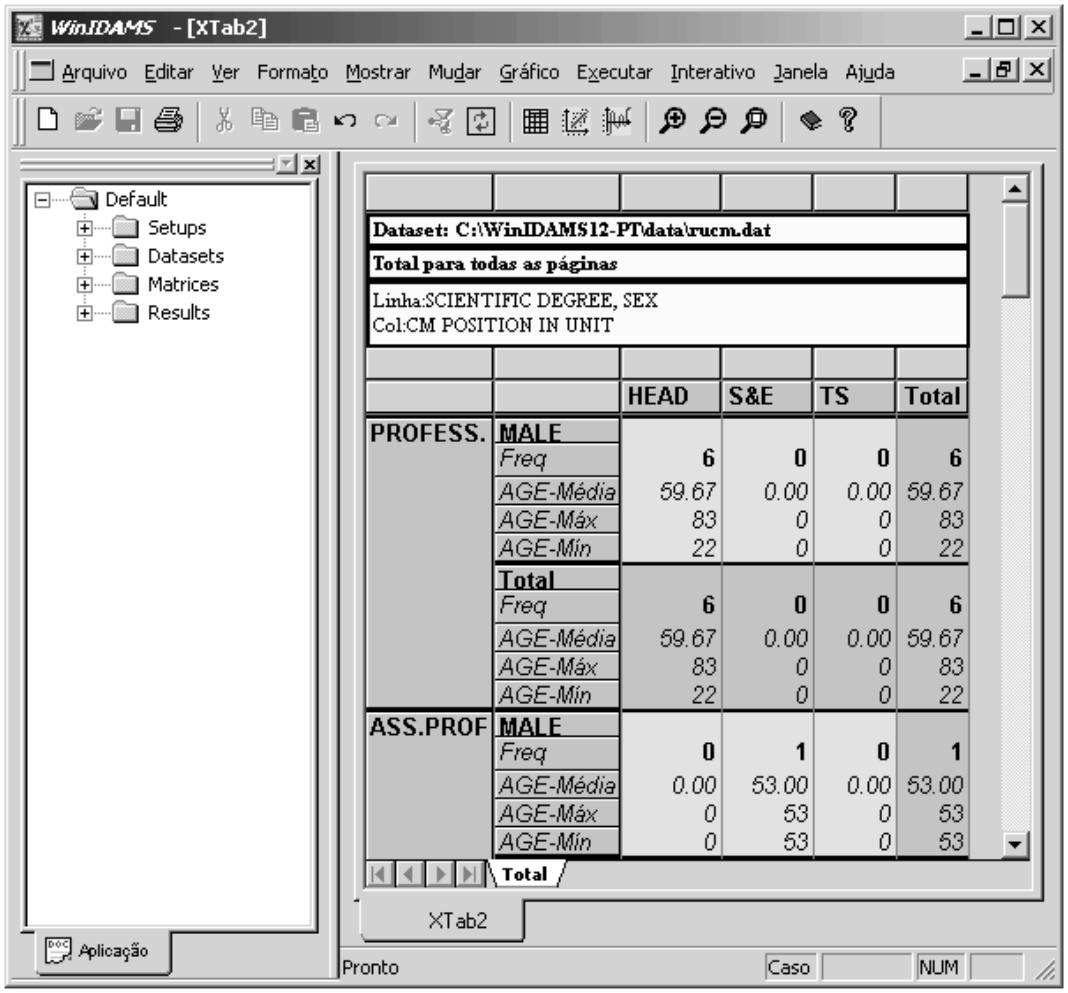
  A ordem das variáveis nas listas VARIÁVEIS DE LINHA e VARIÁVEIS DE COLUNA especifica, implicitamente, a ordem de aninhamento. A primeira variável na lista será a mais externa. A ordem de variável em uma lista pode ser modificada usando a técnica “Drag and Drop”do mouse dentro da mesma lista.



- Depois de selecionar as variáveis, as opções default designadas para uma variável podem ser modificadas através de um clique duplo na variável. Um clique duplo na variável "AGE" na lista VARIÁVEIS DE CÉLULA abre o seguinte diálogo:



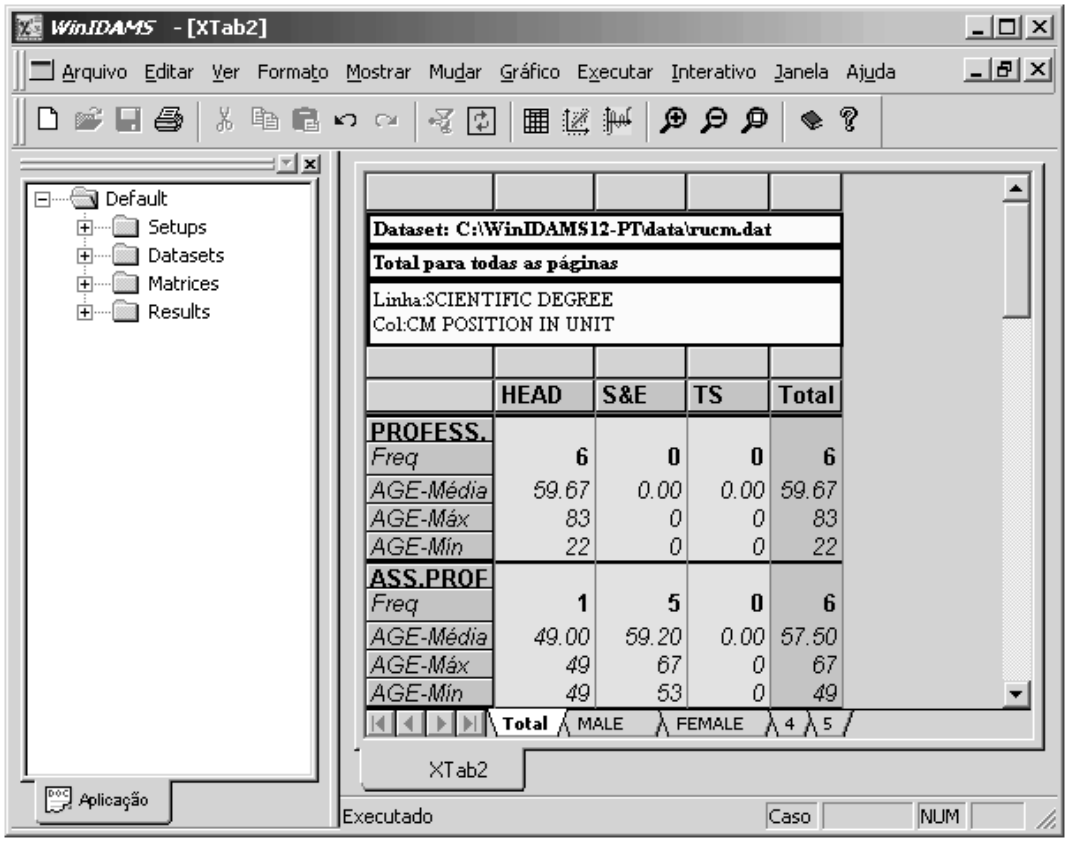
- Média é marcado por default. Marque Máx e Mín. Então clique OK aqui e no diálogo Definição de tabla multidimensional. Você agora vê a tabela multidimensional.



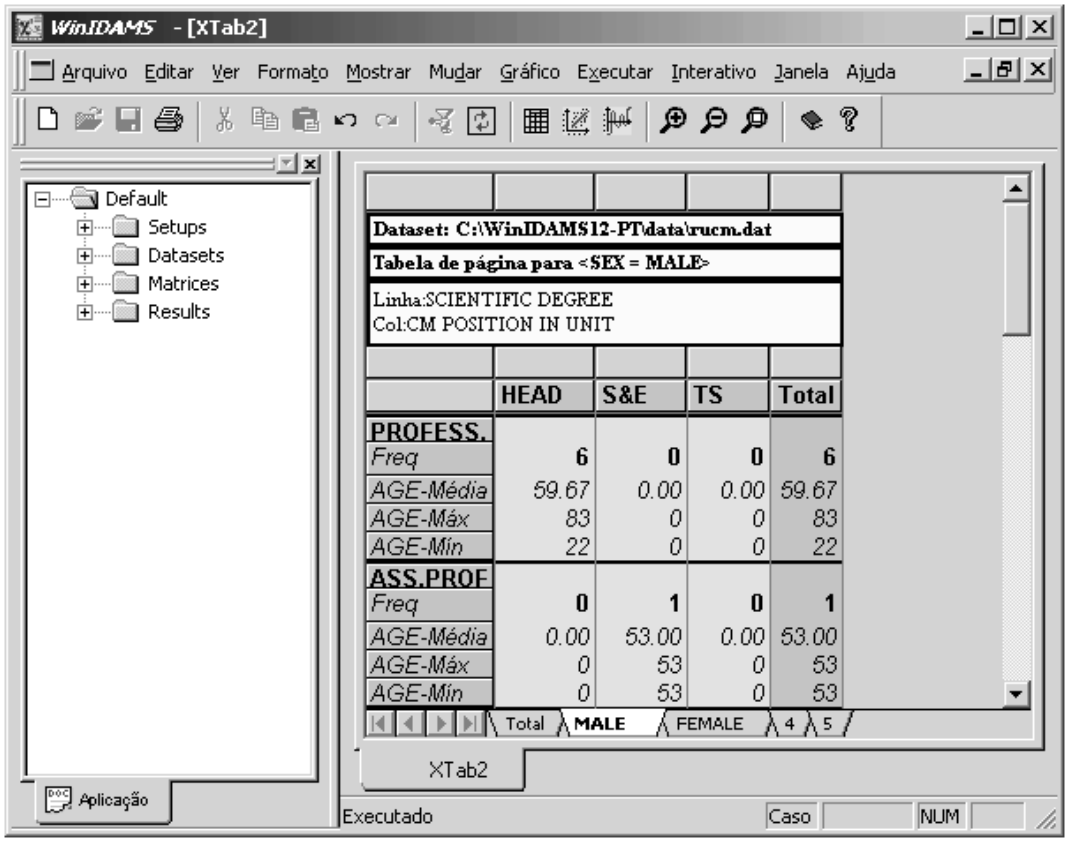
39.6 Como Mudar uma Tabela Multidimensional

Requisitando tabelas separadas. Suponha que agora você deseje ver uma tabela separada para homens e mulheres.

- Clique em Mudar/Especificação e você obtém de volta o diálogo com a seleção prévia de varáveis.
- Use a técnica de "Drag and Drop" para mover a variável "SEX" da lista VARIÁVEIS DE LINHA para a lista VARIÁVEIS DE PÁGINA e clique em OK.
- Você observa a primeira imagem que é o total para todas as variáveis tomadas juntas (homens e mulheres). No final da imagem você pode observar três tabs: "Total", "MALE" e "FEMALE". "Total" é a tab da imagem corrente.



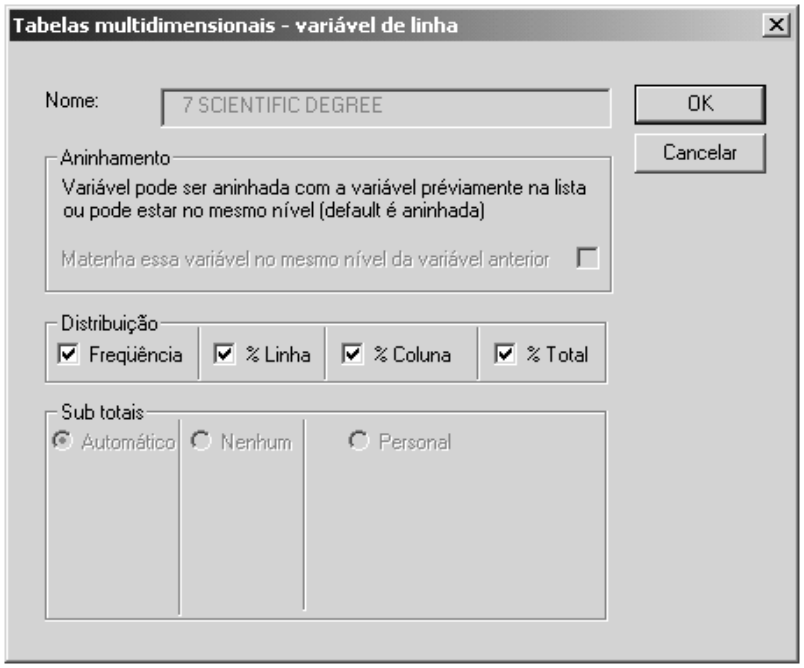
- Para ver a página para os homens, clique na tab "MALE".



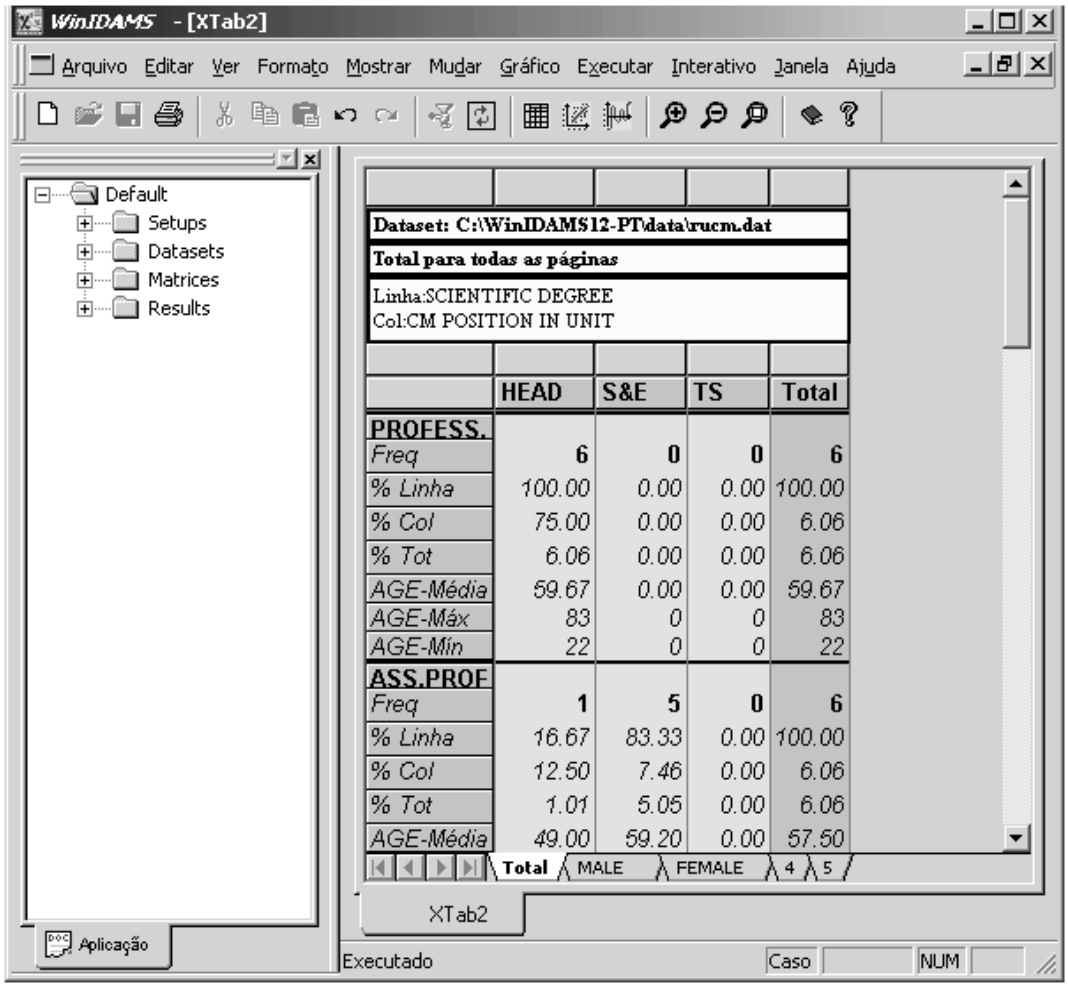
- Para ver a página para os mulheres, clique na tab "FEMALE".

Requisitando percentagens. Enquanto as freqüências são mostradas por default, qualquer tipo de percentagem deve ser requisitada explicitamente.

- Clique em Mudar/Especificação e você obtem de volta o diálogo com a seleção prévia de variáveis.
- Dê um clique duplo na variável de linha "SCIENTIFIC DEGREE"e você vê um diálogo com caixas para Freqüência (marcada por default), % Linha, % Coluna e % Total. Marque todas as caixas de percentagens da seguinte maneira:



- Clique em OK para aceitar essa mudança e clique OK no diálogo Definição de tabla multidimensional. Você vê a tabela multidimensional prévia com todas as percentagens.


WinIDAMS - [XTab2]
Arquivo Editar Ver Formato Mostrar Mudar Gráfico Executar Interativo Janela Ajuda
Default
Setups
Datasets
Matrices
Results
Aplicação
Dataset: C:\WinIDAMS12-PT\data\rucm.dat
Total para todas as páginas
Linha:SCIENTIFIC DEGREE
Col:CM POSITION IN UNIT
HEAD S&E TS Total
PROFESS.
Freq 6 0 0 6
% Linha 100.00 0.00 0.00 100.00
% Col 75.00 0.00 0.00 6.06
% Tot 6.06 0.00 0.00 6.06
AGE-Média 59.67 0.00 0.00 59.67
AGE-Máx 83 0 0 83
AGE-Mín 22 0 0 22
ASS.PROF
Freq 1 5 0 6
% Linha 16.67 83.33 0.00 100.00
% Col 12.50 7.46 0.00 6.06
% Tot 1.01 5.05 0.00 6.06
AGE-Média 49.00 59.20 0.00 57.50
Total MALE FEMALE 4 5
XTab2
Executado
Caso
NUM

Capítulo 40

Exploração Gráfica de Dados

40.1 Visão Geral

GraphID é um componente do WinIDAMS para exploração interativa de dados através de visualização gráfica. Ele aceita dois tipos de entrada:

- Datasets do IDAMS onde os arquivos Dicionário e Dados devem ter o mesmo nome com extensões .dic e .dat respectivamente,
- Arquivos de matriz do IDAMS onde a extensão deve ser .mat.

Somente um dataset ou um arquivo de matriz pode ser usado por vez, i.e. abrindo-se um outro arquivo automaticamente fecha o que estava sendo usado.

40.2 Preparação da Análise

Seleção de dados. Use o comando de menu Arquivo/Abrir ou dê um clique no botão Abrir da barra de ferramentas. Então, na caixa de diálogo Abrir, escolha seu arquivo. Fazendo "Arquivos de tipo:" ser "Arquivos Dados (*.dat)" ou "Arquivos Metriz (*.mat)", permite a filtragem dos arquivos sendo exibidos.

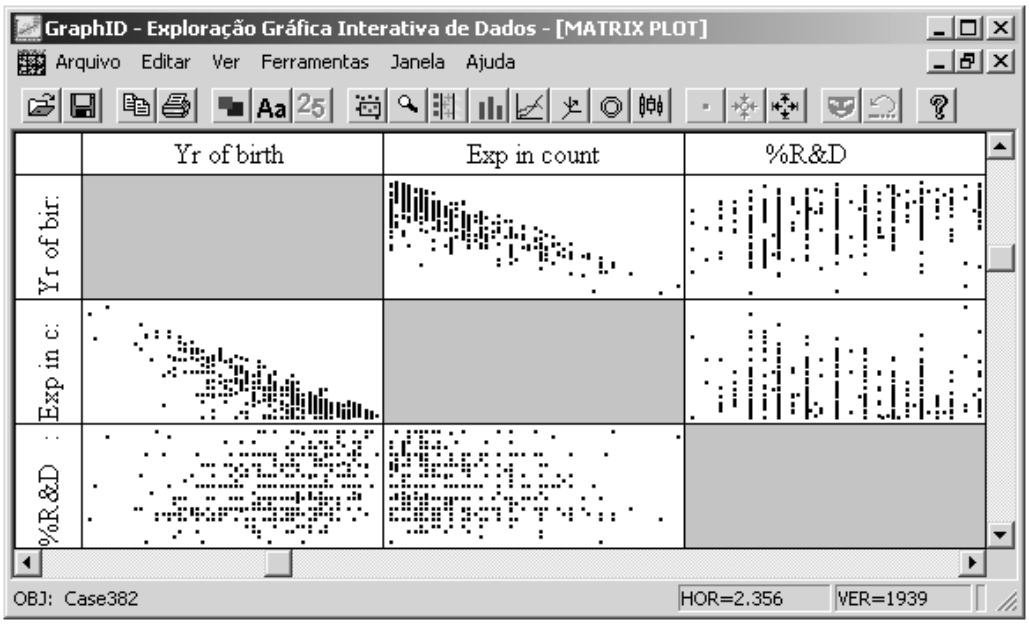
Seleção de identificação de caso. Se você tiver selecionado um dataset, você será requisitado a especificar uma identificação de caso que pode ser uma variável ou número de seqüência do caso. Uma variável numérica ou alfabética pode ser selecionada de uma lista "drop-down".

Seleção de variáveis. Se você tiver selecionado um dataset, você será requisitado a especificar as variáveis que você deseja que sejam analisadas. Variáveis numéricas podem ser selecionadas da "Lista de origem" e movidas para a área "Vars selecionadas". Movendo variáveis entre as listas pode ser feito clicando-se os botões >, < (move somente as variáveis marcadas), >>, << (move todas as variáveis). Note que variáveis alfabéticas não estão disponíveis aqui e que a variável de identificação de caso não é permitida na análise.

Tratamento de dados perdidos. Duas possibilidades são propostas: (1) deleção por casos, quando um caso é usado em análise somente se ele possui dados válidos em todas as variáveis selecionadas; (2) deleção por pares, quando um caso é usado se ele tiver dados válidos em ambas as variáveis para cada par de variáveis separadamente.

40.3 Janela Principal de GraphID para Análise de um Dataset

Depois da seleção de variáveis e um clique em OK, a janela principal GraphID mostra a matriz inicial de diagramas de dispersão com 3 variáveis e as propriedades default da matriz. Essa tela pode ser manipulada usando várias opções e comandos nos menus e/ou ícones equivalentes da barra de ferramentas.



40.3.1 Barra de Menu e Barra de Ferramentas

Arquivo

Abrir	Chama a caixa de diálogo para selecionar um novo dataset/arquivo de matriz para análise.
Fechar	Fecha todas janelas para a análise corrente.
Salvar como	Chama a caixa de diálogo para salvar a imagem gráfica da janela ativa em formato Windows Bitmap (*.bmp).
Salvar casos mascarados	Salva para uso subseqüente, o número seqüencial de casos mascarados durante a sessão, seguindo a seqüência no arquivo Dados analisado.
Imprimir	Chama a caixa de diálogo para imprimir o conteúdo da janela ativa.
Vição preliminar	Exibe uma impressão prévia da imagem gráfica na janela ativa.
Configurar impressora	Chama a caixa de diálogo para modificar as opções de impressão e da impressora.
Sair	Encerra a sessão do GraphID.

O menu pode também conter a lista dos arquivos recentemente abertos, i.e. arquivos usados em sessões prévias do GraphID.

Editar

O menu possui apenas um comando, Copiar, para copiar o gráfico exibido na janela ativa no Clipboard.

Ver

Configuração	Chama a caixa de diálogo para selecionar símbolos, cores, variáveis e o número de colunas e linhas visíveis na matriz.
Escalas	Exibe/oculta as escalas de gráfico para a janela de zoom ativa.
Barra de ferramentas	Exibe/oculta barra de ferramentas.
Barra de status	Exibe/oculta barra de status.
Info	Exibe um janela com informação relevante sobre o dataset: número de casos, número de variáveis, nome do arquivo Dados, etc.

Info de célula	Exibe uma janela com informação relevante sobre a plotagem ativa: nomes de variáveis, seus valores médios, desvios-padrões, correlações e coeficientes de regressão.
Aparência do pincel	Chama a caixa de diálogo para selecionar o símbolo e cor para os casos dentro do pincel.
Fonte para escalas	Chama a caixa de diálogo para selecionar a fonte para escalas para a janela de zoon ativa.
Fonte para nomes	Chama a caixa de diálogo para selecionar a fonte para os nomes das variáveis.
Cores básicas	Chama a caixa de diálogo para selecionar cores para a janela ativa: cor da margem, cor do grid e fundo da célula diagonal.
Salvar cores	Salva modificação de cores.
Salvar fontes	Salva modificação de fontes.

Ferramentas

Nesse menu você pode encontrar ferramentas para manipular a matriz de diagramas de dispersão e para chamar outros gráficos fornecidos por GraphID.

Pincel	Seleciona/cancela a modalidade de pincel.
Zoom	Magnifica a plotagem ativa ou os conteúdos de pincel para a janela completa.
Agrupamento	Chama a caixa de diálogo para especificar a criação de grupos.
Cancelar agrupamento	Cancela o agrupamento.
Histogramas	Chama a caixa de diálogo para especificar gráficos a serem exibidos nas células diagonais e suas propriedades.
Suavização	Chama a caixa de diálogo para especificar tipos de linhas de regressão (linhas de suavização) e suas propriedades.
Diagramas de 3D	Chama a caixa de diálogo para selecionar variáveis a serem usadas como eixos para diagramas de dispersão de 3D e rotação.
Modo direcionado	Seleciona/cancela modo direcionado.
Gráficos de Box-Whisker	Chama a caixa de diálogo para selecionar variáveis e cores para a exibição de gráficos de Box-Whisker.
Jittering	Executa o "jittering" de casos projetados.
Mascarar	Mascara os casos dentro do pincel.
Desmascarar	Restaura, passo a passo, casos mascarados.
Aplicar mascaramento salvo	Mascara os casos que estavam mascarados e salvos na sessão prévia.
Diagrama agrupado	Chama a caixa de diálogo para selecionar variáveis de linha e de coluna para construir tabelas bi-dimensionais, e variáveis X e Y para projetar o diagrama de dispersão dentro das células da tabela.

Janela

O menu contém a lista de janelas abertas e comandos Windows para arranjá-las.

Ajuda

Manual do WinIDAMS	Fornece acesso ao Manual de Referência do WinIDAMS.
Sobre GraphID	Exibe informação sobre a versão e direitos autorais de GraphID e um link para acessar a página na Web do IDAMS no escritório da UNESCO.

Ícones da barra de ferramentas

Há 21 botões na barra de ferramentas fornecendo acesso direto aos mesmos comandos/opções dos menus correspondentes. Eles são listados aqui do jeito que aparecem, da esquerda para a direita.

Abrir	Pincel	Gráficos de Box-Whisker
Salvar	Zoom	Cancelar jittering
Copiar	Agrupamento	Diminuir o nível de jittering
Imprimir	Histogramas	Aumentar o nível de jittering
Cores básicas	Linhas suavizadas	Mascarar casos dentro do pincel
Fonte para nomes	Diagramas de 3D	Restaurar passo a passo casos mascarados
Fonte para escalas	Modo direcionado	Informação sobre GraphID

40.3.2 Manipulação da Matriz de Diagramas de Dispersão

Configurando a matriz de diagramas de dispersão. A matriz corrente de diagramas de dispersão pode ser mudada usando o comando de menu Ver/Configuração.

Visível: Aqui você pode colocar o número de colunas e linhas a serem exibidas na tela (elas não precisam ser iguais). Outras células podem se tornar visíveis através da rolagem de tela.

Variáveis: A caixa de diálogo possui duas listas de variáveis: "Lista de origem" e "Vars selecionadas". Movendo variáveis entre as listas pode ser feito ao se clicar nos botões >, < (move somente variáveis selecionadas), >>, << (move todas as variáveis).

Símbolos: Nessa caixa de diálogo, você pode selecionar a forma e cor dos símbolos que são usados para representar cada grupo de casos nas plotagens. Se nenhum grupo é especificado, então todos os casos pertencem a um grupo único por default e todos serão representados pelo mesmo símbolo (o default é um pequeno retângulo preto). Pode-se designar um símbolo para um grupo ou colapsar grupos ao designar-se o mesmo símbolo para dois ou mais grupos.

A lista de grupos é dada na caixa da esquerda. Duas outras caixas servem para selecionar cores e símbolos. Para selecionar uma cor ou símbolo, basta clicar nela. Sua imagem aparecerá imediatamente no botão próximo ao nome do grupo selecionado.

Modo direcionado. Essa opção é útil quando a ordem dos casos em algumas variáveis de coluna possui um significado, e.g. quando os valores de uma variável de coluna indicam intervalos de tempo. Ligando as imagens seqüencialmente através de linhas retas pode ajudar, por exemplo, a procurar padrões cíclicos.

Para mudar para plotagens direcionadas ou voltar para diagramas de dispersão, pressione o botão Modo direcionado da barra de ferramentas ou use o comando de menu Ferramentas/Modo direcionado.

Mascarando e Desmascarando casos. Você pode mascarar casos projetados em diagramas de dispersão. Essa característica pode ser útil, por exemplo, para remover outliers do gráfico.

Mascarando está disponível quando o pincel está ativo.

Para mascarar casos inclusos dentro pincel, clique o botão Mascarar da barra de ferramentas. Casos mascarados são ocultos em todos diagramas de dispersão. Mascarando pode ser executado várias vezes.

Todos ou parte dos casos mascarados pode ser desmascarados ao se clicar o botão Desmascar da barra de ferramentas.

Salvando e reusando casos mascarados. O número seqüencial dos casos correntemente mascarados pode ser salvo em um arquivo correspondendo ao dataset analisado usando-se o comando Arquivo/Salvar casos mascarados. Essa execução de masking pode ser recuperada em sessões subseqüentes usando o comando Ferramentas/Aplicar mascaramento salvo.

Agrupamento de casos. Essa característica permite você ver como uma variável particiona casos em grupos em todos os diagramas. A variável pode ser tanto quantitativa quanto qualitativa. Além de selecionar

a variável agrupante, o usuário controla a maneira do agrupamento (por valores, ou por intervalos e o número de grupos).

A caixa de diálogo para criação de grupos é ativada ao clicar-se o botãoAgrupamento da barra de ferramentas ou ao usar-se o comando de menu Ferramentas/Agrupamento.

Exploração com o pincel. O pincel é um retângulo que pode ser movido, amplificado e ter sua forma modificada. Ao ser movido ao longo do diagrama de dispersão, os casos dentro do pincel são destacados na cor do pincel e são vistos em todos os outros diagramas de dispersão.

Uma das aplicações é determinar se uma aglomeração de casos em um diagrama de dispersão realmente representa um cluster no espaço ou se a aglomeração é simplesmente uma propriedade da projeção. Para esse propósito, coloque o pincel em uma aglomeração de um diagrama de dispersão e observe como esses casos estão localizados em outros diagramas. Se a mesma aglomeração aparece em outros diagramas, então a aglomeração pode indicar de fato um cluster real. Claro que os diagramas de dispersão devem ser escolhidos de maneira que as distâncias entre casos sejam de mesma magnitude em diagramas diferentes.

Outra aplicação do pincel é a de estudar as distribuições condicionais. Se os 4 cantos do pincel são dados por x_{min}, x_{max}, y_{min}, y_{max}, então os casos dentro do pincel são aqueles que satisfazem as condições:

$$x_{min} < x < x_{max} \quad \text{e} \quad y_{min} < y < y_{max}$$

e os casos satisfazendo essas condições podem ser estudados nos outros diagramas de dispersão.

O pincel pode ser também usado para mascarar e procurar casos.

Para entrar no modo pincel ou cancelá-lo, clique o botão Pincel da barra de ferramentas ou use o comando de menu Ferramentas/Pincel.

Para colocar o pincel na área desejada, coloque o cursor na ponta, pressione o botão da esquerda, puxe e solte na outra ponta.

Para movimentar ou mudar a forma do pincel, coloque o cursor dentro do retângulo do pincel ou a seu lado, pressione o botão esquerdo e puxe. Note: para movê-lo rapidamente para outra célula, coloque o cursor na célula desejada e pressione o botão esquerdo do mouse.

Zooming. Zooming cria uma nova janela para ampliar a célula selecionada ou, em modo pincel, para ampliar o pincel. Essa nova janela com zoon possui a maioria das propriedades de uma matriz de diagramas de dispersão com uma célula; por exemplo, você pode usar o pincel para identificar um novo conjunto de casos e então aplicar o zoon novamente.

Se a matriz parente de diagramas de dispersão está em modo pincel, modificação do pincel é refletida imediatamentena em janela com o zoon; caso contrário, a janela com zoon refletiria modificações introduzidas na célula selecionada da matriz parente.

O comando de menu Ver/Escalas permite a você exibir escalas de valores de variáveis para a janela de zoon ativa.

Jittering. A função é útil quando há variáveis discretas ou qualitativas nos dados analisados. Nesse caso, matrizes usuais de diagramas de dispersão podem não ser muito informativas, pois uma parte de todas as projeções 2D e 3D apresentam grids e, portanto, é impossível determinar visualmente quantos casos coincidem no mesmo grid e a quais grupos eles pertecem.

O jittering é uma transformação dos dados. Valores de dados (x) são modificados ao adicionar-se um "ruído" (a*U) onde U é uma variável aleatória distribuída uniformemente no intervalo (-0.5, 0.5) e a um fator para controlar o nível de jittering.

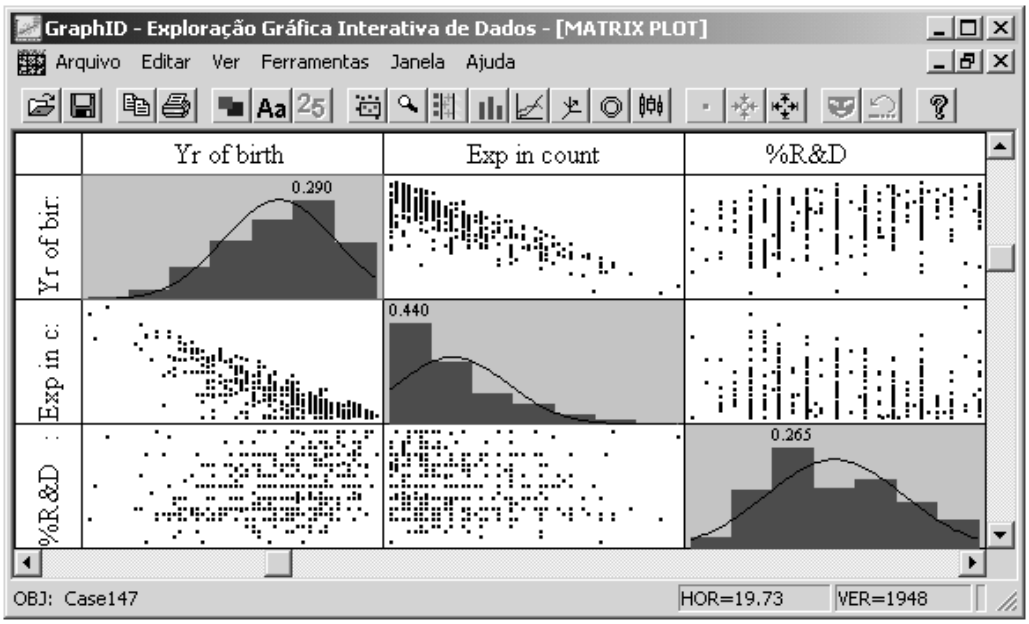
Para colocar o nível desejado de jittering, use os botões da barra de ferramentas: Diminuir o nível de jittering, Aumentar o nível de jittering e Cancelar jittering.

Note que jittering pode ser executado somente na janela da matriz de diagramas de dispersão.

40.3.3 Histogramas e Densidades

Histogramas, densidades normais e gráficos de pontos, e três estatísticas univariadas podem ser exibidas nas células diagonais da matriz de diagramas de dispersão.

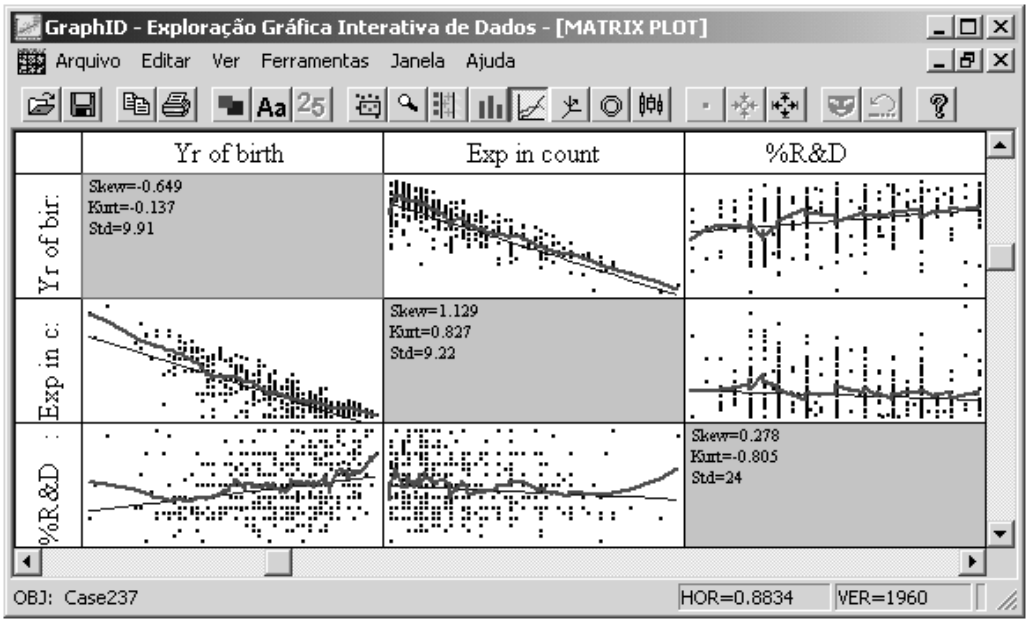
Para obter isso, clique no botão Histogramas da barra de ferramentas ou use o comando de menu Ferramentas/Histogramas. Na caixa de diálogo apresentada, você pode selecionar os gráficos desejados, a cor e o número de barras do histograma. Com a opção Estatísticas, as seguintes estatísticas são fornecidas: Assimetria (Skew), Curtose (Kurt) e Desvio-padrão (Std).



40.3.4 Linhas de Regressão (Linhas suavizadas)

Até 4 linhas de regressão diferentes podem se exibidas em cada diagrama de dispersão:

Regressão linear MLE (estimação de máxima verossimilhança) (regressão linear usual)
Regressão linear local
Média local
Mediana local.



Note que essas são linhas de regressão de Y versus X, onde as variáveis X e Y são projetadas respectivamente no eixo horizontal e vertical.

Para obter as linhas, clique o botão Linhas suavizadas da barra de ferramentas ou use o comando de menu Ferramentas/Suavização. Então, na caixa de diálogo selecione as linhas desejadas, a suas cores e o valor do parâmetro de suavização.

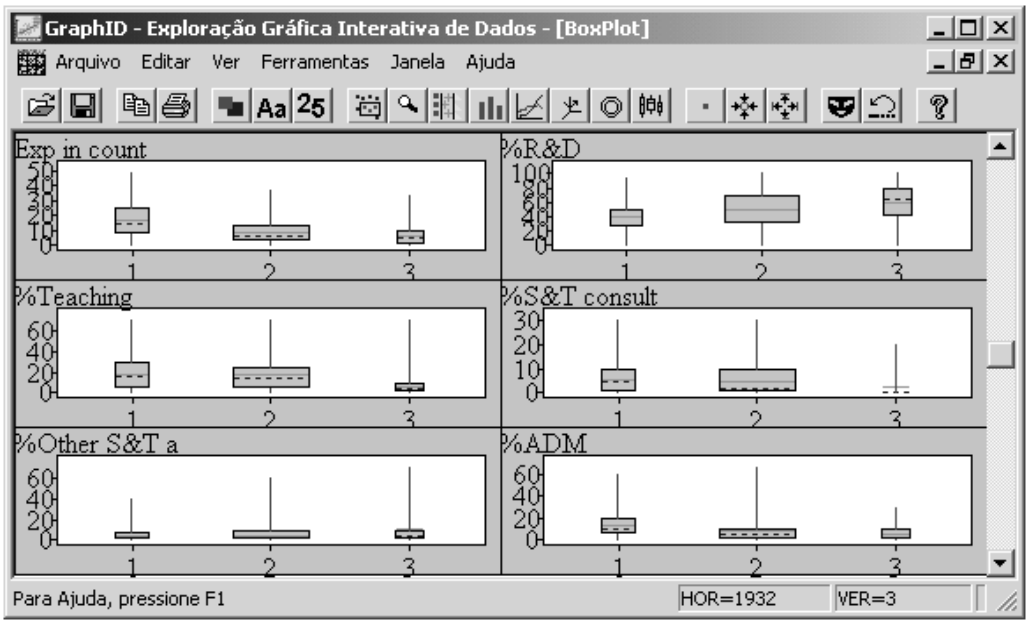
O parâmetro de suavização é o número de vizinhanças. O seu valor default é 7. O valor não pode ser maior que *n/2* onde *n* é o número de casos.

40.3.5 Gráficos de Box-Whisker

Essa característica é especialmente útil se os casos tiverem sido particionados em grupos (ver "Agrupamento de casos" acima).

Use o comando de menu Ferramentas/Gráficos de Box-Whisker ou clique no botão "Gráficos de Box-Whisker" da barra de ferramentas para obter uma caixa de diálogo para especificar o número de colunas e linhas visíveis, como também as cores para a janela das gráficos de Box-Whisker.

Para cada variável selecionada, uma imagem gráfica é exibida na forma de um conjunto de caixas, cada caixa correspondendo a um grupo de casos. A base da caixa pode ser feita proporcional ao número de casos no grupo, e o limite superior e inferior mostram os quantiles superior e inferior, respectivamente. O final superior e inferior das linhas verticais (whiskers) emergindo da caixa correspondem aos valores máximos e mínimos das variáveis para o grupo. As linhas dentro da caixa são a média (linha verde) da variável no grupo e sua mediana (linha azul de hifens). O lado esquerdo do retângulo mostra a escala da variável e sua margem inferior mostra os números dos grupos.



Você pode mudar as cores e fontes dos gráficos usando botões apropriados na barra de ferramentas. Essas mudanças podem ser salvas como novos default para janelas e sessões subseqüentes.

O botão **Cores** permite você mudar as cores de:

Caixas
Fundo
Whiskers
Linha da mediana
Linha da média
Margens.

Os botões **Fonte** permite você modificar as fontes para escalas e nomes de variáveis.

Em qualquer célula de um gráfico de Box-Whisker pode ser usado o zoom. Selecione a célula desejada e clique no botão Zoom da barra de ferramentas.

40.3.6 Diagrama Agrupado

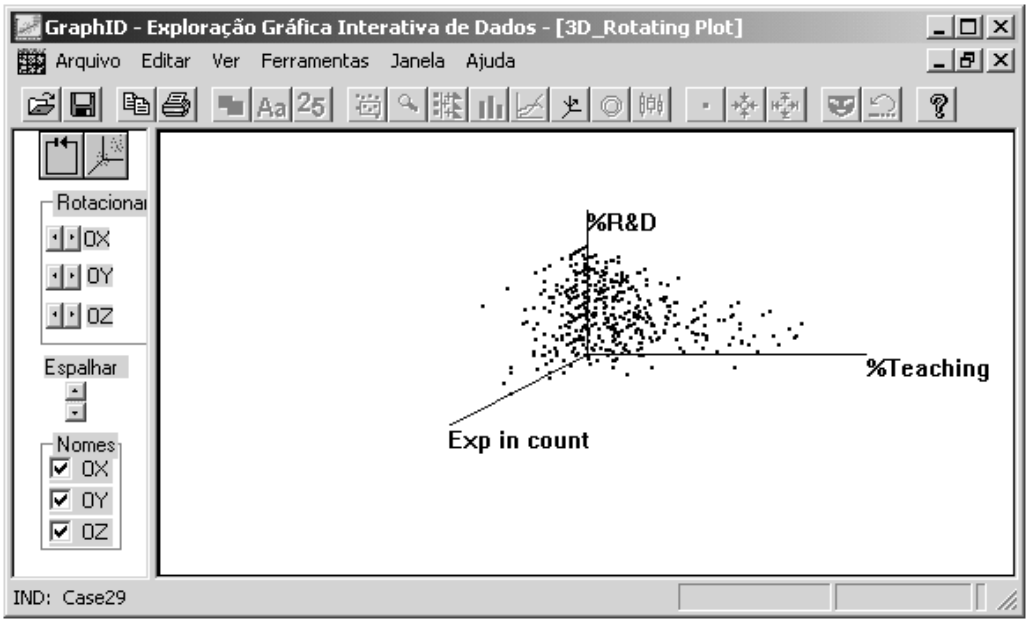
Essa característica permite a projeção de um diagrama de dispersão bi-dimensional dentro de células de uma tabela bi-dimensional, e, portanto, uma análise visual em 4 dimensões.

Use o comando de menu Ferramentas/Diagrama agrupado para obter uma caixa de diálogo para especificar as variáveis de linha e de coluna para a construção da tabela, e variáveis X e Y para os diagramas de dispersão.

Você é requisitado a selecionar a maneira de calcular o número de linhas e colunas. Há duas possibilidades: elas podem ser iguais ao número de valores de variável distintos ou iguais ao número de intervalos especificado pelo usuário. Intervalos calculados possuem o mesmo comprimento.

40.3.7 Diagramas de Dispersão em Três-dimensões e suas Rotações

Para obter um diagrama de dispersão tri-dimensional, clique o botão Diagramas de 3D da barra de ferramentas ou use o comando de menu Ferramentas/Diagramas de 3D. A caixa de diálogo permite que você selecione três variáveis a serem projetadas ao longo dos eixos OX, OY e OZ. Depois de OK, você obtém uma nova janela com um diagrama de dispersão tri-dimensional para as variáveis selecionadas. Se a janela da matriz parente de diagramas está no modo pincel, os casos incluídos no pincel serão exibidos do mesmo jeito desse diagrama.



Você pode usar os elementos de controle da caixa de diálogo no painel esquerdo da janela para modificar a imagem gráfica e rotacioná-la.

O botão no canto esquerdo superior pode ser usado para reajustar os gráficos na posição inicial.

O botão no canto direito superior pode ser usado para ajustar o centro da nuvem de pontos: no centro de gravidade ou no ponto zero.

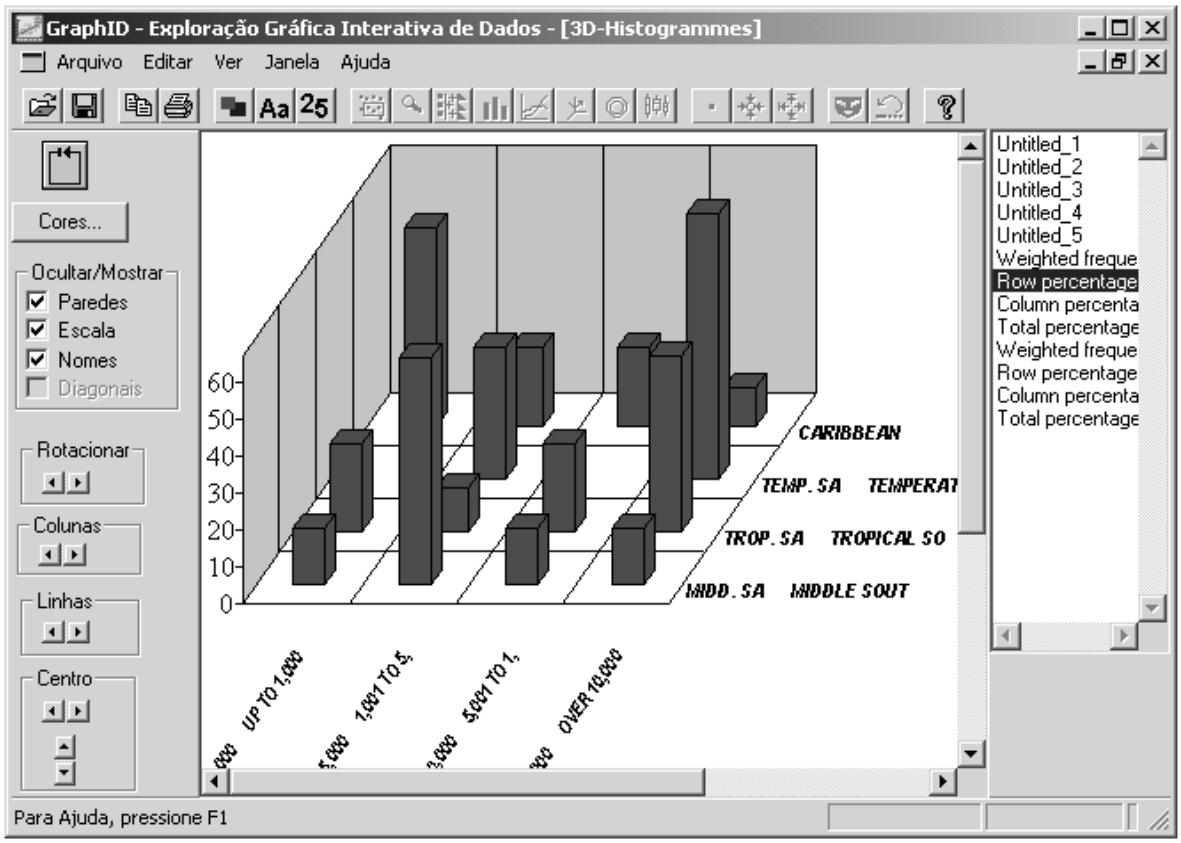
Os botões no grupo **Rotacionar** são usados para rotacionar o diagrama de dispersão ao redor dos eixos correspondentes e aqueles botões no grupo **Espalhar** são usados para mover novos pontos do e para o centro.

O grupo **Nomes** permite você exibir ou esconder os nomes das variáveis nos eixos correpondentes.

Finalmente, o diagrama 3D pode ser projetado como três diagramas de dispersão 2D ao se requisitar 2D-view.

40.4 Janela de GraphID para Análise de uma Matriz

Quando o arquivo com matrizes tiver sido selecionado, você pode clicar em Abrir ou dar um clique duplo no nome do arquivo para exibir um histograma em 3D com uma barra para cada célula da primeira matriz no arquivo. A altura da barra representa o valor da estatística da matriz transformada usando o seu range, i.e. $h = (s_{val} - s_{min})/(s_{max} - s_{min})$. Por default, valores negativos são mostrados em azul e valores positivos em vermelho.



Você pode selecionar cores para nomes e escalas, valores positivos e negativos, paredes, piso e fundo. Use a mesma técnica usada em gráficos de Box-Whisker.

Na parte direita da janela, você é apresentado a uma lista de matrizes incluídas no arquivo. Note que somente os primeiros 16 caracteres da descrição do conteúdo da matriz são exibidos. Se não houver descrição, GraphID exibe "Untitled_n". Você pode exibir a matriz desejada clicando na descrição do seu conteúdo.

A exibição da matriz pode ser manipulada usando opções e comandos nos itens da barra de menu e/ou ícones equivalentes na barra de ferramentas.

40.4.1 Barra de Menu e Barra de Ferramentas

Arquivo e Editar

Os mesmos comandos dos menus correspondentes na análise de dataset, exceto Fechar, são fornecidos.

Ver

Barra de ferramentas	Exibe/oculta a barra de ferramentas.
Barra de status	Exibe/oculta a barra de status.
Cores	Chama a caixa de diálogo para selecionar cores para a janela ativa: nomes de linha/coluna e escalas, valores positivos e negativos, paredes, piso e fundo.
Fonte para escalas	Chama a caixa de diálogo para selecionar a fonte para as escalas.
Fonte para nomes	Chama a caixa de diálogo para selecionar a fonte para os nomes.

Janela e Ajuda

Os mesmos comandos dos menus correspondentes na análise de dataset estão disponíveis.

Ícones da barra de ferramentas

Botões estão disponíveis na barra de ferramentas oferecendo acesso direto aos mesmos comandos/opções dos menus correspondentes. Eles são listados aqui do modo que aparecem da esquerda para a direita.

Abrir
Salvar
Copiar
Imprimir
Cores
Fonte para nomes
Fonte para escalas
Informação sobre GraphID.

40.4.2 Manipulação da Matriz Exibida

Similarmente à manipulação de diagramas de dispersão 3D, você pode usar os elementos de controle da caixa de diálogo no painel esquerdo da janela para mudar a imagem gráfica e para rotacionar a matriz exibida.

O botão superior pode ser usado para reajustar o gráfico na sua posição inicial.

O botão **Cores** permite que você mude as cores de:

Barra (valores positivos)
Parede
Barra (valores negativos)
Piso
Fundo
Nomes e escala.

Caixas do grupo **Ocultar/Mostrar** permitem a você exibir ou ocultar paredes, escala, nomes nos eixos correspondentes e a diagonal, se aplicável.

Os botões no grupo **Rotacionar** podem ser usados para rotacionar a matriz ao longo do eixo vertical.

Os botões nos grupos **Colunas** e **Linhas** podem ser usados para mudar o tamanho das colunas e linhas respectivamente.

Os botões no grupo **Centro** permite você mover-seno gráfico para a esquerda, direita, para cima e para baixo.

Capítulo 41

Análise de Séries Temporais

41.1 Visão Geral

TimeSID é um componente do WinIDAMS para análise de séries temporais. Ele usa datasets do IDAMS como entrada onde os arquivos de dicionário e de dados devem possuir a mesma extensão .dic e .dac, respectivamente.

Somente um dataset pode ser usado por vez, i.e. a abertura de outro dataset automaticamente fecha aquele sendo utilizado.

41.2 Preparação da Análise

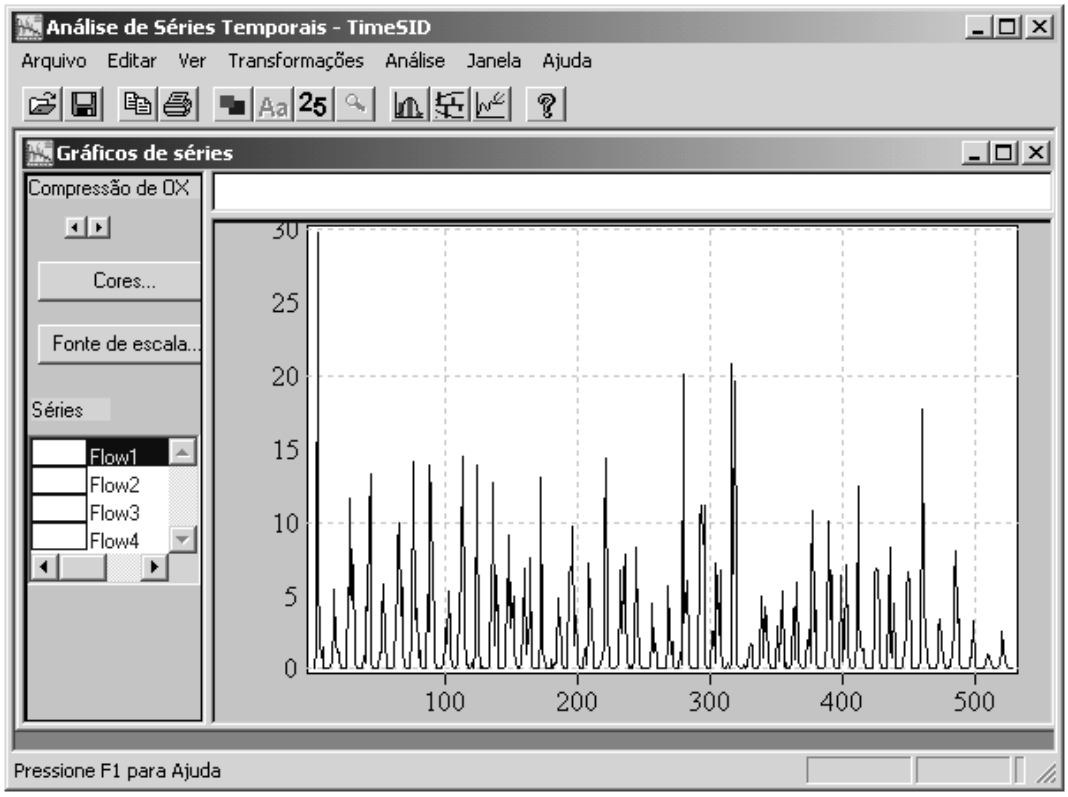
Seleção de dados. Use o comando de menu Arquivo/Abrir ou clique o botão Abrir da barra de ferramentas. Então, na caixa de diálogo Abrir, selecione seu arquivo. Fazendo "Arquivos de tipo:" igual a "Arquivos Dados (*.dat)" exibirá apenas arquivos de dados do IDAMS.

Seleção de séries. Você será requisitado a especifcar as séries (variáveis) que você quer analisar. Variáveis numéricas podem ser selecionadas da lista "Séries acessíveis" e movidas para a área "Séries selecionadas". Mover variáveis entre listas pode ser feito clicando-se nos botões >, < (move somente variáveis selecionadas), >>, << (move todas as variáveis). Note que variáveis alfabéticas não estão disponíveis aqui.

Tratamento de dados perdidos. Valores de dados perdidos são excluídos das transformações das séries; eles são também excluídos do cálculo das estatísticas e autocorrelações. Para as outras análises, valores de dados perdidos são substituídos pela média geral.

41.3 Janela Principal de TimeSID

Depois da seleção de variáveis e um clique no OK, a janela principal de TimeSID exibe o gráfico da primeira série da lista de séries selecionadas. As séries podem ser manipuladas e analisadas usando várias opções e comandos nos menus e/ou ícones equivalentes da barra de ferramentas.



41.3.1 Barra de Menu e Barra de Ferramentas

Arquivo

Abrir	Chama a caixa de diálogo para selecionar um novo dataset para análise.
Fechar	Fecha todas janelas para a análise corrente.
Salvar como	Chama a caixa de diálogo para salvar os conteúdos da janela/painel ativo. Imagens gráficas são salvas em formato de Windows Bitmap format (*.bmp). Tabela de dados e tabelas com estatísticas são salvas em formato de texto.
Imprimir	Chama a caixa de diálogo para imprimir os conteúdos da janela/painel ativo.
Vição preliminar	Exibe uma impressão prévia dos conteúdos da janela/painel ativo.
Configurar impressora	Chama a caixa de diálogo para modificar as opções de impressão e da impressora.
Sair	Encerra a sessão do TimeSID.

O menu pode conter também a lista de arquivos recentemente abertos, i.e. arquivos usados em sessões prévias do TimeSID.

Editar

O menu possui um comando, Copiar, para copiar os conteúdos da janela/painel ativo para o Clipboard.

Ver

Barra de ferramentas	Exibe/oculta barra de ferramentas.
Barra de status	Exibe/oculta barra de status.
Escala OX	Exibe/oculta a escala OX para a série temporal.
Fonte para escalas	Chama a caixa de diálogo para selecionar a fonte para as escalas.
Cores básicas	Chama a caixa de diálogo para selecionar cores para a margem e fundo.

Janela

Tabela de dados	Chama a janela com a tabela de dados. Colunas da tabela de dados são as série de tempo analisadas (incluindo os resultados de transformação).

Além de Tabela de dados, o menu contém a lista de janelas abertas e opções Windows para arranjá-las.

Ajuda

Manual do WinIDAMS	Propicia acesso ao Manual de Referência do WinIDAMS.
Sobre TimeSID	Mostra a informação sobre a versão e direito de propriedade de TimeSID e um link para acessar a página na Web do IDAMS no escritório da UNESCO.

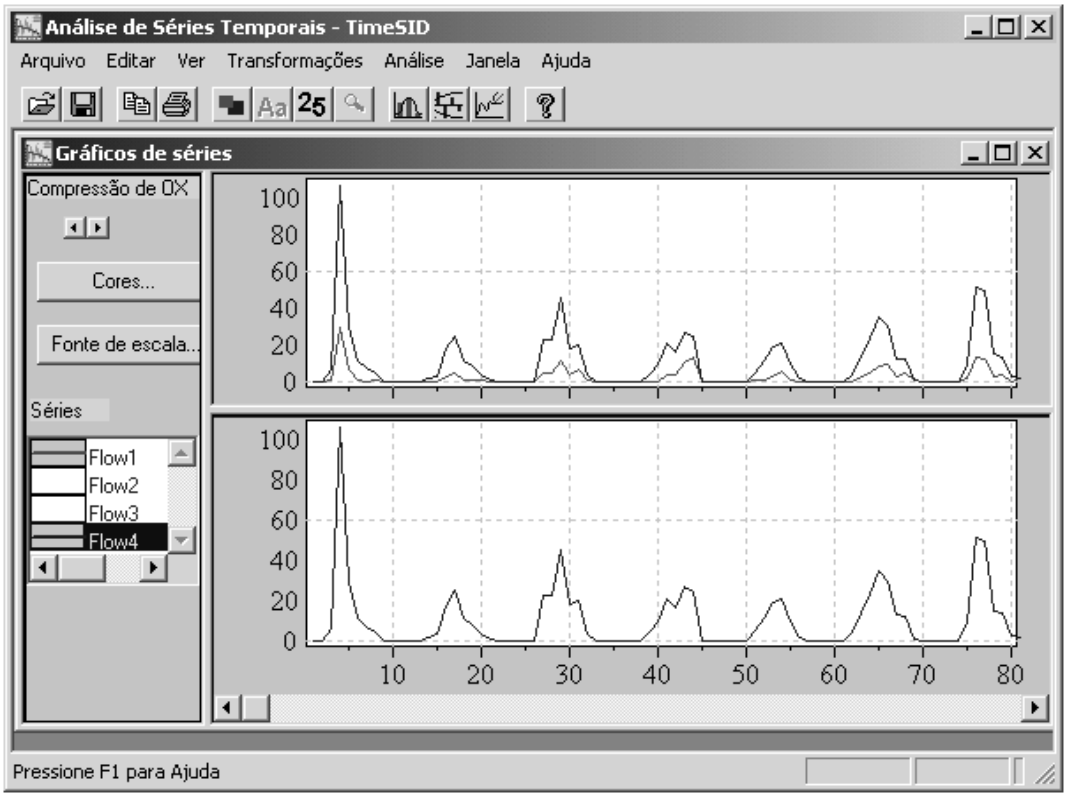
Os dois outros menus, Transformações e Análise, são descritos em detalhes nas seções "Transformação de Séries Temporais" e "Análise de Séries Temporais" abaixo.

Ícones de barra de ferramentas

Existem 9 botões ativos na barra de ferramenta propiciando acesso direto aos mesmos comandos/opções que os itens de menu correspondentes. Eles são listados aqui como eles aparecem da esquerda para a direita.

Abrir	Histogramas, características estatísticas básicas
Copiar	Autocorrelações e correlações cruzadas
Imprimir	Autoregressão
Cores básicas	Informação sobre TimeSID
Fonte para escalas	

41.3.2 A Janela de Séries Temporais



A janela de séries temporais é dividida em três painéis: o da esquerda é para mudar as propriedades e para selecionar séries (variáveis), o da direita superior é para exibir várias séries temporais e o da direita inferior é para exibir a série atual.

Mudando a aparência do painel. Os dois painéis para exibir séries temporais são sincronizados e eles podem ser mudados usando os controles fornecidos no painel da esquerda. Por default, o painel direito superior está vazio e o seu tamanho é reduzido. O painel direito inferior exibe as séries correntes, mantendo as barras de rolagem e escalas visíveis. O tamanho de cada painel pode ser mudado usando o mouse, e a escala OX pode ser ocultada/exibida usando o comando Escala OX do menu Ver. Além do mais, apresentação de gráficos pode ser modificada da seguinte maneira:

- *regulação do grau de compressão do gráfico* - use os botões sob Compressão de OX,
- *cores para pano de fundo e margens* - use o botão Cores ou comando Ver/Cores básicas,
- *fonte para escalas* - use o botão Fonte para escalas ou o comando Ver/Fonte para escalas.

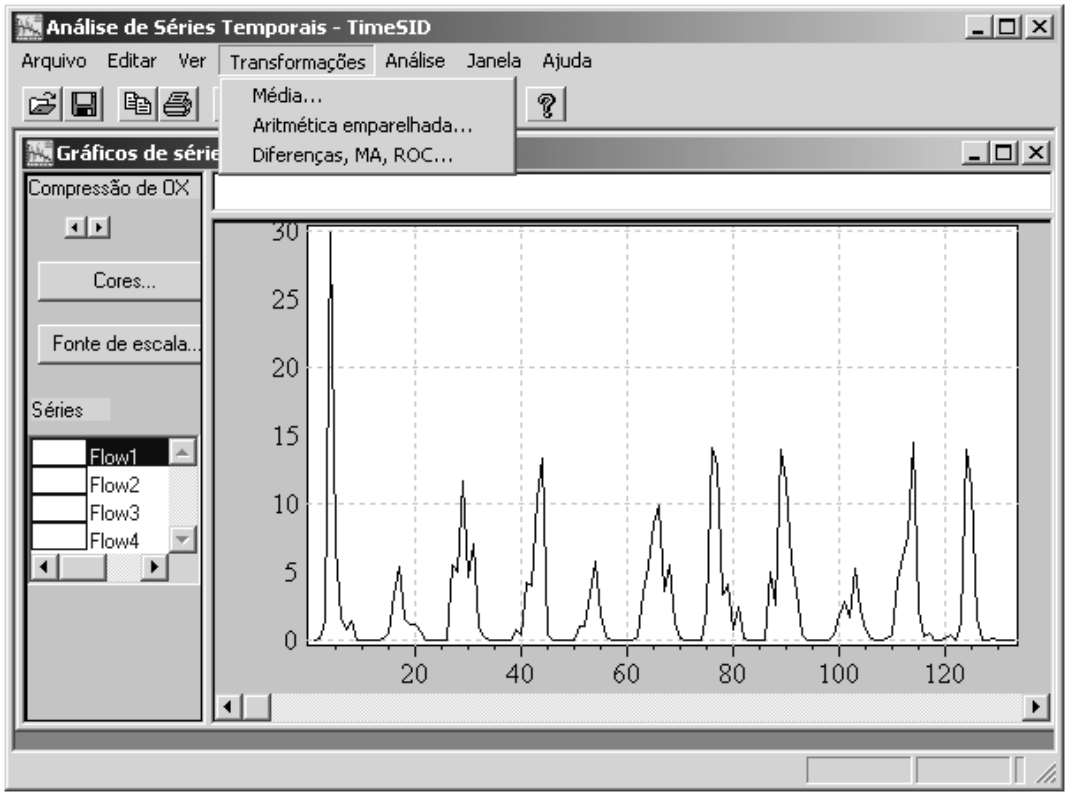
Mudando o nome das séries temporais. Selecione a série temporal requisitada, clique no seu nome com o botão direito do mouse e selecione a opção Mudar nome. A janela ativa apresenta o nome para modificação. Note que essas modificações são temporárias e que elas são mantidas somente durante a sessão corrente.

Selecionando séries temporais para exibição. Uma lista de séries temporais analisadas é fornecida no painel esquerdo. Dando um clique duplo em uma variável na lista, você pode escolher a forma e a cor da linha de projeção. Depois de OK, o gráfico correspondente é exibido no painel superior. Essa operação pode ser repetida para diferentes variáveis e, portanto, pode-se obter vários gráficos exibidos simultaneamente no painel superior. O painel direito inferior exibe as séries correntes.

Deletando séries temporais da análise. Selecione as séries temporais requisitadas, clique em seus nomes com o botão direito do mouse e selecione a opção Deleter série.

41.4 Transformação de Séries Temporais

Dados de séries temporais podem ser transformados pelo cálculo de diferenças, alisamento, supressão de tendência, usando um número de funções, etc. O menu **Transformações** contém comandos para criar novas séries temporais baseadas em valores das séries selecionadas. Note que variáveis exibidas para seleção são renumeradas seqüencialmente começando de zero (0).



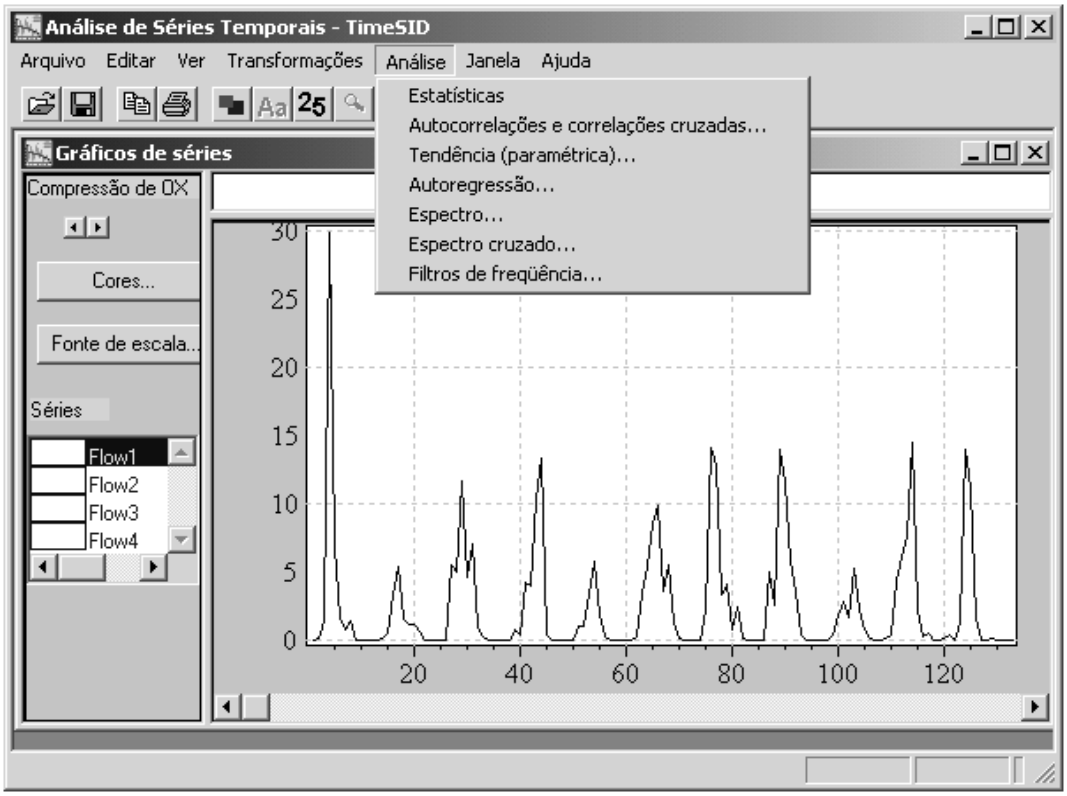
Média cria uma nova série temporal como uma média das séries especificadas. Séries a serem tomadas para o cálculo são selecionadas na caixa de diálogo "Seleção de séries" (ver seção "Preparação da Análise").

Aritmética emparelhada cria um conjunto de séries temporais pela execução de operações aritméticas em pares de séries temporais especificadas na caixa de diálogo (cada série especificada na primeira lista de argumentos com o segundo argumento).

Diferenças, MA, ROC cria um conjunto de séries temporais baseado em transformações (diferenças seqüenciais, média móvel não centrada, taxa de mudança) das séries especificadas na caixa de diálogo. Parâmetros específicos para cada transformação como também o tipo de transformação ROC são ajustados na mesma caixa de diálogo.

41.5 Análise de Séries Temporais

Características de análises são ativadas através de comandos no menu **Análise.**



Estatísticas cria uma tabela com média, desvio-padrão, valores mínimo e máximo, como também a tabela com estatísticas para testar a hipóteses "randomicidade versus tendência" para a séries temporal selecionada. Ele também exibe um histograma para essa série.

Autocorrelações e correlações cruzadas cria uma nova janela com um conjunto de células contendo gráficos de autocorrelações e correlações cruzadas para o conjunto de séries temporais especificadas.

Tendência (paramétrica) cria uma nova série temporal como produto da estimação de um modelo paramétrico de tendência para séries temporais especificadas. O modelo de tendência e as séries são selecionadas em uma caixa de diálogo.

Autoregressão estimativas de um modelo de autoregressão para previsão de curto prazo para as séries temporais especificadas.

Espectro (análise espectral) cria uma tabela de valores de espectro (freqüência, período, densidade), gráfico de estimação de espectro, e para espectro DFT, gráfico de desvios do espectro cumulativo em relação ao espectro cumulativo de um "ruído branco". Ele pode usar a transformação de Fourier discreta rápida (DFT) e/ou o método de entropia máxima (MENT) para estimação da densidade. No procedimento

DFT, duas janelas são utilizadas para obter a estimativa melhorada da densidade espectral: janela de dados Welch no domínio de tempo e alisamento polinomial no domínio de freqüência.

Espectro cruzado analisa um par de séries estacionárias. Ele fornece os valores de potência espectral cruzada, função de fase e coerência, como também as suos gráficos. O espectro cruzado é estimado usando a janela de alisamento de Parzen.

Filtros de freqüência decompõe uma série temporal em componentes de freqüência. Eles criam uma nova série ao aplicar um dos seguintes filtros: freqüência baixa, freqüência alta, band-pass ou band-cut. Para filtro de freqüência alta ou baixa, o seu limite freqüência é igual ao valor do parâmetro de *Freqüência*. Para filtro de band-pass ou band-cut, os limites de freqüência são determinados pelo intervalo *(Freqüência - Largura de janela, Freqüência + Largura de janela)*. Uma opção Eliminar tendência permite a retirada da tendência de uma série temporal antes da filtragem (o componente de tendência é adicionado aos resultados da filtragem).

Referências

Farnum, N.R., Stanton, L.W., *Quantitative Forecasting Methods*, PWS-KENT Publishing Company, Boston, 1989.

Kendall, M.G., Stuart, A., *The Advanced Theory of Statistics*, Volume 3 - Design and Analysis, and time series, Second edition, Griffin, London, 1968.

Marple Jr, S.L., *Digital Spectral Analysis with Applications*, Prentice-Hall, Inc., 1987.

Parte VI

Fórmulas Estatísticas e Referências Bibliográficas

Capítulo 42

Análise de Agrupamento

Notação

x	=	valores das variáveis
h, i, j, l	=	subscritos dos objetos
f, g	=	subscritos das variáveis
p	=	número de variáveis
c	=	subscrito para o cluster
k	=	número de clusters
N_j	=	número de objetos no cluster j
N	=	número total de casos.

42.1 Estatísticas Univariadas

Se a antrada for um dataset do IDAMS, as seguintes estatísticas são calculadas para todas as variáveis utilizadas na análise:

a) ***Média.***

$$\overline{x}_f = \frac{\sum_i x_{if}}{N}$$

b) ***Desvio-médio absoluto.***

$$s_f = \frac{\sum_i |x_{if} - \overline{x}_f|}{N}$$

42.2 Medidas Padronizadas

Na mesma situação, o programa pode computar medidas padronizadas, também chamadas z-scores, dadas por:

$$z_{if} = \frac{x_{if} - \overline{x}_f}{s_f}$$

para cada caso i e cada variável f usando o valor médio e o desvio-médio absoluto da variável f (ver seção 1 acima).

42.3 Matriz de Dissimilaridade Computada de um Dataset do IDAMS

Os elementos d_{ij} de uma matriz de dissimilaridade medem o grau de dissimilaridade entre casos i e j. Os d_{ij} são calculados diretamente dos dados originais, ou dos z-scores se as variáveis são requisitadas a serem padronizadas. Uma das duas distâncias pode ser escolhida: euclidiana ou city-block.

a) ***Distância euclidiana.***

$$d_{ij} = \sqrt{\sum_{f=1}^{p}(x_{if} - x_{jf})^2}$$

b) ***Distância city-block.***

$$d_{ij} = \sum_{f=1}^{p}|x_{if} - x_{jf}|$$

42.4 Matriz de Dissimilaridade Computada de uma Matriz de Similaridade

Se a entrada consiste de uma matriz de similaridade com elementos s_{ij}, os elementos d_{ij} da matriz de dissimilaridade são calculados da seguinte maneira:

$$d_{ij} = 1 - s_{ij}$$

42.5 Matriz de Dissimilaridade Computada de uma Matriz de Correlação

Se a entrada consiste de uma matriz de correlação com elementos r_{ij}, os elementos d_{ij} da matriz de dissimilaridade são calculados usando uma das duas fórmulas: SIGN ou ABSOLUTE.

Ao se usar a fórmula SIGN, variáveis com uma correlação positiva alta recebem um coeficiente de dissimilaridade próximo a zero, de outro modo, variáveis com uma correlação negativa forte serão consideradas muito dissimilares.

$$d_{ij} = (1 - r_{ij})/2$$

Ao usar a fórmula ABSOLUTE, variáveis com uma correlação negativa ou positiva alta receberão uma pequena dissimilaridade.

$$d_{ij} = 1 - |r_{ij}|$$

42.6 Partição ao Redor de Medoids (PAM)

O algoritmo busca k objetos representativos (medoides) que estão centralmente localizados nos clusters que eles definem. O objeto representativo de um cluster, o medoide, é o objeto para o qual a dissimilaridade média de todos os objetos no cluster é mínima. De fato, o algoritmo PAM minimiza a soma de dissimilaridades ao invés da dissimilaridade média.

A seleção de k medoides é executada em duas fases. Na primeira fase, um agrupamento inicial é obtido pela sucessiva seleção de objetos representativos até que k objetos tenham sido encontrados. O primeiro objeto é aquele para o qual a soma das dissimilaridades em relação a todos os outros objetos é a menor possível.

(Isso é um tipo de "mediana multivariada" dos N objetos, por isso o termo "medoide".) Subseqüentemente, a cada passo, PAM seleciona o objeto que diminui a função objetivo (soma de dissimilaridades) tanto quanto possível. Na segunda fase, uma tentativa é feita para melhorar o conjunto de objetos representativos. Isso é feito considerando-se todos os pares de objetos (i, h) cujo objeto i foi selecionado e objeto h não, checando se selecionando h e deselecionando i reduz a função objetivo. Em cada passo, a troca mais econômica é mantida.

a) ***Distância média final (dissimilaridade).*** Essa é a função objetivo do PAM, que pode ser visto como uma medida de "adequação" do agrupamento.

$$\text{Distância média final} = \frac{\sum_{i=1}^{N} d_{i,m(i)}}{N}$$

onde $m(i)$ é o objeto representativo (medoide) mais próximo do objeto i.

b) ***Clusters isolados.*** Há dois tipos de clusters isolados: L-clusters e L^*-clusters.

Cluster $\mathcal{C}$ é um L-cluster se para cada objeto i pertencendo a $\mathcal{C}$

$$\max_{j \in \mathcal{C}} d_{ij} < \min_{h \notin \mathcal{C}} d_{ih}$$

Cluster $\mathcal{C}$ é um L^*-cluster se

$$\max_{i,j \in \mathcal{C}} d_{ij} < \min_{l \in \mathcal{C}, h \notin \mathcal{C}} d_{lh}$$

c) ***Diâmetro de um cluster.*** O diâmetro do cluster $\mathcal{C}$ é definido como a maior dissimilaridade entre objetos pertencentes a $\mathcal{C}$:

$$\text{Diâmetro}_{\mathcal{C}} = \max_{i,j \in \mathcal{C}} d_{ij}$$

d) ***Separação de um cluster.*** A separação do cluster $\mathcal{C}$ é dfinida como a menor dissimilaridade entre dois objetos, um dos quais pertence ao cluster $\mathcal{C}$ e o outro não.

$$\text{Separação}_{\mathcal{C}} = \min_{l \in \mathcal{C}, h \notin \mathcal{C}} d_{lh}$$

e) ***Distância média a um medoide.*** Se j é o medoide do cluster $\mathcal{C}$, a distância média de todos os objetos de $\mathcal{C}$ em relação a j é calculada da seguinte maneira:

$$\text{Distância média}_j = \frac{\sum_{i \in \mathcal{C}} d_{ij}}{N_j}$$

f) ***Distância máxima a um medoide.*** Se o objeto j é o medoide do cluster $\mathcal{C}$, a distância máxima de todos os objetos de $\mathcal{C}$ em relação a j é calculada da seguinte maneira:

$$\text{Distância máxima}_j = \max_{i \in \mathcal{C}} d_{ij}$$

g) ***Silhuetas de cluster.*** Cada cluster é representado por uma silhueta (Rousseeuw 1987), mostrando que objetos se posicionam bem dentro do cluster e quais meramente ficam em uma posição intermediária. Para cada objeto, a seguinte informação é fornecida:

- o número de clusters ao qual ele pertence (CLU),
- o número do cluster vizinho (NEIG),
- o valor s_i (denotado por S(I) no resultados),
- o identificador de três-caracteres do objeto i,
- uma linha, cujo comprimento é proporcional a s_i.

Para cada objeto i o valor s_i é calculado da seguinte maneira:

$$s_i = \frac{b_i \ - \ a_i}{\max(a_i, \ b_i)}$$

onde a_i é a dissimilaridade média do objeto i em relação a todos os outros objetos do cluster $\mathcal{A}$, que contém i e onde b_i é a dissimilaridade média do objeto i em relação a todos os outros objetos do cluster mais próximo $\mathcal{B}$ (vizinho do objeto i). Note que o cluster vizinho é um tipo de segundo-melhor para o objeto i. Quando o cluster $\mathcal{A}$ contém apenas um objeto i, o s_i é zero ($s_i = 0$).

h) ***Largura média de silhueta de um cluster.*** É a média de s_i para todos os objetos i em um cluster.

i) ***Largura média de silhueta.*** É a média de s_i para todos os objetos i nos dados, i.e. largura média de silhueta para k clusters. Isso pode ser utilizado para selecionar o "melhor" número de clusters, escolhendo aquele k dando a maior média de s_i.

Outro coeficiente, SC, chamado COEFICIENTE DE SILHUETA, pode ser calculado manualmente como a largura média máxima de silhueta ao longo de todo o k para o qual a silhueta pode ser construída. Esse coeficiente é uma medida adimensional da quantidade de estrutura de agrupamento que foi descoberta pelo algoritmo de classificação.

$$\text{SC} = \max_k \ \overline{s}_k$$

Rousseeuw (1987) propôs a seguinte interpretação do coeficiente SC:

$0.71 - 1.00$	Uma estrutura forte foi encontrada.
$0.51 - 0.70$	Uma estrutura razoável foi encontrada.
$0.26 - 0.50$	A estrutura é fraca e pode ser artificial; por favor, tente métodos adicionais nesses dados.
≤ 0.25	Nenhuma estrutura substancial foi encontrada.

42.7 Agrupamento Aplicado a Grandes Volumenes de Dados (CLARA)

Similarmente a PAM, o método CLARA é também baseado na busca por k objetos representativos. Mas o algoritmo CLARA é desenhado especialmente para analisar grandes conjuntos de dados. Conseqüentemente, a entrada de CLARA deve ser um dataset do IDAMS.

Internamente, CLARA conduz dois passos. Primeiro uma amostra é coletada do conjunto de objetos (casos), e dividida em k clusters usando o mesmo algoritmo de PAM. Então, cada objeto não pertecendo a amostra é designado para o mais próximo objeto representativo, em relação aos k objetos. A qualidade desse agrupamento é definida como a distância média entre cada objeto e seu objeto representativo. Cinco dessas amostras são coletadas e depois submetidas a um cluster e, então, aquela com a menor distância média obtida é selecionada.

O agrupamento retido do conjunto de dados inteiro é, então, analisado mais profundamente. A distância final média, as distâncias média e máxima em relação a cada medoide são calculadas do mesmo jeito como em PAM (para todos os objetos, e não apenas aqueles selecionados na amostra). Silhuetas de clusters e estatísticas relacionadas são também calculadas do mesmo jeito que em PAM, mas apenas para objetos na amostra selecionada (pois o gráfico da silhueta completa seria muito grade para imprimir).

42.8 Agrupamento Difuso (FANNY)

Agrupamento difuso é uma generalização do particionamento, que pode ser aplicada ao mesmo tipo de dado que o método PAM, mas o algoritmo é de natureza diferente. Ao invés de designar um objeto para um cluster particular, FANNY dá o seu grau de "belonging" (coeficiente de filiação) para cada cluster, e, portanto, propicia informação muito mais detalhada da estrutura dos dados.

a) ***Função objetivo.*** A técnica de agrupamento difuso usada em FANNY pretende minimizar a função objetivo

$$\text{Função objetivo} = \sum_{c=1}^{k} \frac{\sum_i \sum_j u_{ic}^2 \, u_{jc}^2 \, d_{ij}}{2 \sum_j u_{jc}^2}$$

onde u_{ic} e u_{jc} são funções de filiação que estão sujeitas às restrições

$$u_{ic} \geq 0 \qquad \text{para} \;\; i = 1, 2, \ldots, N; \; c = 1, 2, \ldots, k$$

$$\sum_c u_{ic} = 1 \quad \text{para} \;\; i = 1, 2, \ldots, N$$

O algoritmo minimizando essa função objetivo é iterativo e pára quando a função converge.

b) ***Agrupamento difuso (filiações).*** Esses são os valores de filiação (coeficiente de filiação u_{ic}) que fornecem o menor valor da função objetivo. Eles indicam, para cada objeto i, quão intensamente ele pertence ao cluster c. Note que a soma dos coeficientes de filiação é igual a 1 para cada objeto.

c) ***Coeficiente de partição de Dunn.*** Esse coeficiente, F_k, mede quão "duro" um agrupamento difuso é. Ele varia de um mínimo de $1/k$ para um agrupamento completamente difuso (onde todos $u_{ic} = 1/k$) até um valor de 1 para um agrupamento inteiramente "duro" (onde todos $u_{ic} = 0$ ou 1).

$$F_k = \sum_{i=1}^{N} \sum_{c=1}^{k} u_{ic}^2 \, / \, N$$

d) ***Coeficiente de partição normalizado de Dunn.*** A versão normalizada do coeficiente de partição de Dunn sempre varia de 0 até 1, seja qual for o valor de k escolhido.

$$F_k' = \frac{F_k \; - \; (1/k)}{1 \; - \; (1/k)} = \frac{kF_k \; - \; 1}{k \; - \; 1}$$

e) ***Agrupamento duro mais próximo.*** Essa partição (= agrupamento "duro") é obtida ao se designar cada objeto ao cluster no qual ele possui o maior coeficiente de filiação. Siluetas de clusters e estatísticas relacionadas são calculadas da mesma maneira que em PAM.

42.9 Agrupamento Hierárquico Aglomerativo (AGNES)

Esse método pode ser aplicado ao mesmo tipo de dados que os dos métodos PAN e FANNY. Contudo, não é mais preciso especificar o número de clusters requeridos. O algoritmo constrói uma hierarquia do tipo árvore que contém, implicitamente, todos os valores de k, iniciando com N clusters e procedendo por meio de fusões sucessivas até que um único cluster seja obtido com todos os objetos.

No primeiro passo, os dois objetos mais próximos (i.e. com a menor dissimilaridade inter-objeto) são juntos para constituir um cluster com dois objetos, enquanto os outros clusters mantêm apenas um membro. Em cada passo sucessivo, os clusters mais próximos (com a menor dissimilaridade inter-objeto) são fundidos.

a) ***Dissimilaridade entre dois clusters.*** No algoritmo AGNES, o método de média de grupo de Sokal e Michener (às vezes chamado "método da média de grupo-emparelhado não-ponderado") é usado para medir dissimilaridades entre clusters.

Faça $\mathcal{R}$ e $\mathcal{Q}$ denotar dois clusters e $|\mathcal{R}|$ e $|\mathcal{Q}|$ denotar seus números de objetos. A dissimilaridade $d(\mathcal{R}, \mathcal{Q})$ entre clusters $\mathcal{R}$ and $\mathcal{Q}$ é definida como a média de todas as dissimilaridades d_{ij}, onde i é qualquer objeto de $\mathcal{R}$ e j é qualquer objeto de $\mathcal{Q}$.

$$d(\mathcal{R}, \mathcal{Q}) = \frac{1}{|\mathcal{R}| \; |\mathcal{Q}|} \sum_{i \in \mathcal{R}} \sum_{j \in \mathcal{Q}} d_{ij}$$

b) ***Ordenamento final de objetos e dissimilaridades entre eles.*** Na primeira linha, os objetos são listados na ordem em que eles aparecem na representação gráfica dos resultados. Na segunda linha, as dissimilaridades entre clusters que se juntam são impressas. Note que o número de dissimilaridades impressas é um a menos que o número de objetos N, porque há $N-1$ fusões.

c) ***Banner de dissimilaridades.*** É uma representação gráfica dos resultados. Um banner consiste de estrelas e listas. As estrelas indicam as ligações e as linhas são repetições de identificadores de objetos. Um banner é sempre lido da esquerda para a direita. Cada linha com estrelas se inicia na dissimilaridade entre os clusters sendo fundidos. Existem escalas fixas acima e abaixo do banner, indo de 0.00 (dissimilaridade 0) a 1.00 (maior dissimilaridade encontrada). A maior dissimilaridade de fato (correspondendo a 1.00 no banner) é fornecida logo abaixo do banner.

d) ***Coeficiente aglomerativo.*** A largura média do banner é chamada de coeficiente aglomerativo (AC). Ele descreve a intensidade da estrututra de agrupamento que foi encontrada.

$$AC = \frac{1}{N}\sum_i l_i$$

onde l_i é o comprimento da linha contendo o identificador do objeto i.

42.10 Agrupamento Hierárquico Divisivo (DIANA)

O método DIANA pode ser usado para os mesmos tipos de dados como no método AGNES. Apesar de AGNES e DIANA produzirem um output similar, DIANA constói a sua hierarquia na direção oposta, começando com um grande cluster contendo todos os objetos. A cada passo, ele divide um cluster em dois clusters menores, até que todos os clusters contenham apenas um único elemento. Isso significa que para N objetos, a hierarquia é construída em $N-1$ passos.

No primeiro passo, os dados são separados em dois clusters fazendo-se uso das dissimilaridades. Em cada passo subseqüente, o cluster com o maior diâmetro (ver 6.c acima) é dividido da mesma maneira. Depois de $N-1$ passos divisivos, todos os objetos estarão separados.

a) ***Dissimilaridade média em relação a todos os outros objetos.*** Faça $\mathcal{A}$ denotar um cluster e $|\mathcal{A}|$ denotar seu número de objetos. A dissimilaridade média entre o objeto i e todos os outros objetos no cluster $\mathcal{A}$ é definida como em 6.g acima.

$$d_i = \frac{1}{|\mathcal{A}|-1}\sum_{j\in\mathcal{A}, j\neq i} d_{ij}$$

b) ***Ordenamento final de objetos e diâmetros dos clusters.*** Na primeira linha, os objetos são listados na ordem em que eles aparecem na representação gráfica. Os diâmetros dos clusteres são impresso logo em baixo. Essas duas seqüências de números juntas caracterizam a hierarquia completa. O maior diâmetro indica o nível no qual o conjunto de dados completos é dividido. Os objetos a esquerda desses valores constituem um cluster, e os objetos no lado direito constituem um outro cluster. O segundo maior diâmetro indica a segunda divisão, e assim sucessivamente.

c) ***Banner de dissimilaridades.*** Em relação ao método AGNES, trata-se de uma representação gráfica dos resultados. Ele também consiste de linhas de estrelas, e das listras que repetem os identificadores dos objetos. O banner é lido da esquerda para direita mas as escalas fixas acima e abaixo do banner variam agora de 1.00 (correspondendo ao diâmetro do conjunto de dados completo) e 0.00 (correspondendo ao diâmetro dos singletons). Cada linha com estrelas termina no diâmetro onde o cluster é dividido. O diâmetro real do conjunto de dados (correspondendo a 1.00 no banner) é fornecido logo abaixo do banner.

d) ***Coeficiente divisivo.*** A largura média do banner é chamada de coeficiente divisivo (DC). Ele descreve a intensidade da estrutura de cluster encontrada.

$$DC = \frac{1}{N}\sum_i l_i$$

onde l_i é o comprimento da linha contendo o identificador do objeto i.

42.11 Agrupamento Monotético (MONA)

O método MONA é destinado a dados que consistam exclusivamente de variáveis binárias (dicotômicas) (aquelas que podem assumir apenas dois valores, e portanto $x_{if} = 0$ ou $x_{if} = 1$). Apesar do algoritmo ser do tipo divisivo hierárquico, ele não usa dissimilaridades entre objetos, e portanto, a matriz de dissimilaridade não é computada. A divisão entre clusters usa as variáveis diretamente.

A cada passo, uma das variáveis (digamos, f) é utilizada para dividir os dados pela separação de objetos i, para os quais $x_{if} = 1$ daqueles onde $x_{if} = 0$. No próximo passo, cada cluster obtido no passo anterior é novamente dividido, usando valores (0 e 1) de uma das variáveis remanescentes (diferentes variáveis podem ser usadas em diferentes clusters). O processo é continuado até que cada cluster contenha apenas um objeto, ou até que as variáveis remanescentes não possam separá-lo.

Para cada divisão, a variável mais fortemente associada com as outras variáveis é escolhida.

a) ***Associação entre duas variáveis.*** A medida de associação entre duas variáveis f e g é definida pelo seguinte:

$$A_{fg} = |a_{fg}d_{fg} - b_{fg}c_{fg}|$$

onde a_{fg} é o número de objetos i com $x_{if} = x_{ig} = 0$, d_{fg} é o número de objetos com $x_{if} = x_{ig} = 1$, b_{fg} é o número de objetos com $x_{if} = 0$ e $x_{ig} = 1$, e c_{fg} é o número de objetos com $x_{if} = 1$ e $x_{ig} = 0$.

A medida A_{fg} expressa se as variáveis f e g fornecem divisões similares do conjunto de objetos, e pode ser considerada como um tipo de similaridade entre variáveis.

Para selecionar a variável mais fortemente associada com outras variáveis, a medida total A_f é calculada para cada variável f da seguinte maneira:

$$A_f = \sum_{g \neq f} A_{fg}$$

b) ***Ordenamento final de objetos.*** Os objetos são listados na ordem em que eles aparecem no gráfico de separação (banner). Os passos de separação e as variáveis utilizadas para separação são impressas abaixo de identificadores de objetos.

c) ***Gráfico de separação (banner).*** Essa representação gráfica é bastante similar ao banner impressa por DIANA. O comprimento de uma linha de estrelas é agora proporcional ao número do passo onde a separação foi conduzida. Linhas de identificadores de objetos correspondem a objetos. Uma linha de identificadores que não continue no lado direito do banner sinaliza um objeto que se tornou um cluster singleton naquele passo correspondente. Linhas de identificadores plotados entre duas linhas de estrelas indicam objetos que pertencem a um cluster que não pode ser separado.

42.12 Referências

Kaufman, L., and Rousseeuw, P.J., *Finding Groups in Data: An Introduction to Cluster Analysis*, John Wiley & Sons, Inc., New York, 1990.

Rousseeuw, P.J., Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis, *Journal of Computational and Applied Mathematics*, 20, 1987.

Capítulo 43

Análise de Configuração

Notação

Seja $\mathcal{A}_{(n,t)}$ uma matriz retangular de n variáveis (linhas) e t dimensões (colunas). Uma variável ou ponto a posui t coordenadas, cada uma correspondendo a uma dimensão.

$$\begin{aligned} a_{is} &= \text{elemento da matriz } \mathcal{A} \text{ na linha } i \text{ e coluna } s \\ i, j &= \text{subscritos para variáveis(linhas)} \\ n &= \text{número de variáveis} \\ s, l, m &= \text{subscritos para dimensões(columns)} \\ t &= \text{número de dimensões.} \end{aligned}$$

43.1 Configuração Centrada

As variáveis são centradas dentro de cada dimensão pela subtração da média de cada coluna de cada elemento na coluna.

$$\text{Centrado } a_{is} = a_{is} - \frac{\sum_i a_{is}}{n}$$

Depois da aplicação dessa fórmula, a média das coordenadas das n variáveis é zero para cada dimensão.

43.2 Configuração Normalizada

A soma dos quadrados de todos os elementos da matriz $\mathcal{A}$ dividido pelo número de variáveis n dá a média dos segundos momentos das variáveis. Cada elemento da matriz é normalizado pela raiz quadrada desse valor.

$$\text{Normalizado } a_{is} = \frac{a_{is}}{\sqrt{\sum_i \sum_s a_{is}^2 / n}}$$

Depois dessa normalização, a soma dos quadrados dos elementos a_{is} é igual a n.

43.3 Solução com Eixos Principais

A configuração é rotacionada de maneira que as dimensões sucessivas sejam responsáveis pela maior variância possível. Seja $\mathcal{A}$ a configuração a ser rotacionada e $\mathcal{B}$ a configuração na forma de eixos principais.

Cálculo da matriz $\mathcal{B}$:

A matriz simétrica $\mathcal{A}'\mathcal{A}$ de dimensões (t,t) é computada primeiro. Então, os vetores próprios, $\mathcal{T}$, de $\mathcal{A}'\mathcal{A}$ são determinados usando o método de diagonalização de Jacobi.

A matriz $\mathcal{A}$ é transformada em uma matriz $\mathcal{B}$ de elementos, tal que $\mathcal{B} = \mathcal{A}\mathcal{T}$, $\mathcal{B}$ possuem n linhas e t colunas como a matriz $\mathcal{A}$.

43.4 Matriz de Produtos Escalares

$$\mathrm{SP}_{ij} = \sum_s a_{is}\, a_{js}$$

A matriz SP de dimensões (n,n) é uma matriz quadrada e simétrica de produtos escalares de variáveis. O produto escalar de uma variável por ela mesma é o seu segundo momento. Se cada variável é centrada e normalizada (média = 0, desvio-padrão = 1), a matriz SP se torna a matriz de correlação.

43.5 Matriz de Distâncias Entre Pontos

$$\mathrm{DIST}_{ij} = \sqrt{\sum_s (a_{is} - a_{js})^2}$$

DIST é uma matriz quadrada e simétrica de distâncias euclidianas entre variáveis.

43.6 Configuração Rotacionada

A rotação pode ser executada somente em duas dimensões por vez. É função do usuário selecionar as dimensões, e.g. 2 e 5 (coluna 2 e coluna 5) e o ângulo ϕ de rotação em termos de graus.

Novas coordenadas são calculadas da seguinte maneira:

$$\begin{aligned} a'_{il} &= a_{il}\cos\phi + a_{im}\sin\phi \\ a'_{im} &= -a_{il}\sin\phi + a_{im}\cos\phi \end{aligned}$$

O cálculo é executado para cada valor de i, e tantas vezes quantas ainda existirem variáveis.

Na matriz $\mathcal{A}$, as colunas l e m se tornam os vetores das novas coordenadas calculadas como indicado acima.

43.7 Configuração Translada

A translação pode ser executada apenas em uma única dimensão (uma coluna) por vez. O usuário especifica a constante T a ser adicionada a cada elemento da dimensão, e a coluna l onde isso será aplicado.

Para todas as coordenadas de l (n coordenadas, dado que há n variáveis):

$$a'_{il} = a_{il} + T$$

43.8 Rotação Varimax

(a) Os elementos a_{is} de $\mathcal{A}$ são normalizados pela raiz quadrada das comunidades correspondentes a cada variável, e se define

$$b_{is} = \frac{a_{is}}{\sqrt{\sum_s a_{is}^2}}$$

(b) Tendo construído $\mathcal{B} = (b_{is})$, procura-se pelos melhores eixos de projeção para as variáveis, depois da equalização de suas inércias. A maximização da função V_c é feita através de rotações sucessivas de duas dimensões por vez, até a convergência ser atingida.

$$V_c = \sum_s \frac{n \sum_i b_{is}^4 - \left(\sum_i b_{is}\right)^2}{n^2}$$

A matriz resultante $\mathcal{B}$ de elementos b_{is} possui o mesmo número de linhas e colunas que a matriz inicial $\mathcal{A}$.

43.9 Configuração Classificada

Essa é a configuração final impressa em um formato diferente. Cada dimensão é impressa como uma linha, com os elementos para as dimensões em ordem crescente.

43.10 Referências

Greenstadt, J., The determination of the characteristic roots of a matrix by the Jacobi method, *Mathematical Methods for Digital Computers*, eds. A. Ralston and H.S. Wilf, Wiley, New York, 1960.

Herman, H.H., *Modern Factor Analysis*, University of Chicago Press, Chicago, 1967.

Kaiser, H.F., Computer program for varimax rotation in factor analysis, *Educational and Psychological Measurement*, 3, 1959.

Capítulo 44

Análise Discriminante

Notação

x	=	valores das variáveis
k	=	subscrito para caso
i, j	=	subscritos para variáveis
g	=	superescrito para grupo
q	=	subscrito para passo
p	=	número de variáveis
w	=	valor do peso
x_k^g	=	vetor de p elementos correspondendo ao caso k no grupo g
y_q^g	=	vetor valores médios das variáveis selecionadas no passo q para o grupo g
N^g	=	número de casos no grupo g
W^g	=	soma total de pesos para o grupo g
I_q	=	subconjunto de índices para variáveis selecionadas no passo q.

44.1 Estatísticas Univariadas

Essas estatísticas, ponderadas se os pesos forem especificados, são calculadas para cada grupo e para cada variável de análise, usando a amostra básica. A média é calculada também para a amostra básica total (média total).

a) ***Média.***

$$\overline{x}_i^g = \frac{\sum_{k=1}^{N^g} w_k^g \, x_{ki}^g}{W^g}$$

Nota: a média total é calculada usando uma fórmula análoga.

b) ***Desvio-padrão.***

$$s_i^g = \sqrt{\frac{\sum_{k=1}^{N^g} w_k^g \left(x_{ki}^g\right)^2}{W^g} - \left(\overline{x}_i^g\right)^2}$$

44.2 Discriminação Linear Entre 2 Grupos

O procedimento é baseado na função linear discriminante de Fisher e usa a matriz de covariância total para calcular os coeficientes dessa função. Classificação de casos é feita usando-se os valores dessa função, e não distâncias. O critério aplicado para selecionar-se a variável seguinte é o D^2 de Mahalanobis (distância de Mahalanobis entre dois grupos). Depois de cada passo, o programa fornece a função discriminante linear, a tabela de classificação e o percentual de casos corretamente classificados para ambas as amostras, i.e., básica e de teste.

a) ***Função discriminante linear.*** Vamos denotar a função calculada no passo q como

$$f_q(x) = \sum_{i \in I_q} b_{qi}\, x_i \; + a_q$$

Os coeficientes b_{qi} dessa função para as variáveis i inclusas no passo q correspondem aos elementos do único valor próprio da matriz

$$(y_q^1 - y_q^2)' \; \mathcal{T}_q^{-1}$$

e o termo constante é calculado da seguinte maneira:

$$a_q = -\frac{1}{2}(y_q^1 - y_q^2)' \; \mathcal{T}_q^{-1} \; (y_q^1 + y_q^2)$$

onde $\mathcal{T}_q$ é a matriz de covariância total (calculada para os casos oriundos dos dois grupos) para as variáveis incluídas no passo q, com elementos

$$t_{ij} = \frac{\sum_k w_k \, (x_{ki} - \overline{x}_i)(x_{kj} - \overline{x}_j)}{W^1 + W^2}$$

b) ***Tabela de classificação para amostra básica.***

Um caso é designado:

para o grupo 1 se $f_q(x) > 0$,
para o grupo 2 se $f_q(x) < 0$.

Um caso não é designado se $f_q(x) = 0$.

Percentual de casos classificados corretamente é calculado como o quociente entre o número de casos na diagonal e o número de casos totais na tabela de classificação.

c) ***Tabela de classificação para amostra de teste.***

Construída da mesma maneira que na amostra básica (ver 2.b acima).

d) ***Critério para selecionar a variável seguinte.*** A distância de Mahalanobis entre dois grupos é usada para essa finalidade. A variável selecionada no passo q é aquela que maximiza o valor de D_q^2.

$$D_q^2 = (y_q^1 - y_q^2)' \; \mathcal{T}_q^{-1} \; (y_q^1 - y_q^2)$$

e) ***Alocação e valor da função discriminante linear para os casos.*** Esses são calculados e impressos para o último passo, ou quando o passo precede uma queda do percentual de casos corretamente classificados. O valor da função é calculado de acordo com a fórmula descrita abaixo do ponto 2.a acima; as variáveis utilizadas no cálculo são aquelas retidas no passo. A designação de casos para os grupos é feita como descrito no ponto 2.b acima.

A mesma fórmula e regras de designação são usadas para a amostra básica, as médias de grupo, a amostra de teste e a amostra anônima.

44.3 Discriminação Linear Entre Mais Do Que 2 Grupos

O procedimento para discriminação de 3 ou mais grupos usa não somente a matriz de covariância total mas também a matriz de covariância entre grupos. O critério para selecionar a variável seguinte usado aqui é o traço de um produto dessas duas matrizes (generalização da distância de Mahalanobis para dois grupos). Depois de selecionar a nova variável a ser introduzida, a análise de fator discriminante é executada e o programa fornece a potência discriminante total e a potência discriminante para os três primeiros fatores. Casos são classificados de acordo com suas distâncias do centro dos grupos. Em cada passo, o programa calcula e imprime a tabela de classificação e o percentual de casos corretamente classificados em ambas as amostras (básica e de teste).

a) ***Tabela de classificação para amostra básica.*** A distância de um caso x do centro de um grupo g no passo q é definida como a função linear

$$v_{y_q^g}(x) = (y_q^g)' \, \mathcal{T}_q^{-1} \, (y_q^g - 2x)$$

onde $\mathcal{T}_q$, como descrito em 2.a acima, é a matriz de covariância total (calculada para os casos de todos os grupos) para as variáveis inclusas no passo q, com os elementos

$$t_{ij} = \frac{\sum_k w_k \, (x_{ki} - \overline{x}_i)(x_{kj} - \overline{x}_j)}{W}$$

Um caso é designado para o grupo cujo $v_{y_q^g}(x)$ possua o menor valor (a menor distância).

O PERCENTUAL DE CASOS CLASSIFICADOS CORRETAMENTE é calculado como o quociente entre o número de casos na diagonal e o número total de casos na tabela de classificação.

b) ***Tabela de classificação para amostra de teste.***

Construído do mesmo jeito da tabela de amostra básica (ver 3.a acima).

c) ***Critério para seleção da variável seguinte.*** A variável selecionada no passo q é aquela que maximiza o valor do traço da matriz $\mathcal{T}_q^{-1} \, \mathcal{B}_q$, onde $\mathcal{T}_q$ é a matriz de covariância total usada no passo q (ver 3.a acima), e $\mathcal{B}_q$ é a matriz de covariância entre grupos, com elementos

$$b_{ij} = \frac{\sum_g W^g \, (y_i^g - \overline{x}_i)(y_j^g - \overline{x}_j)}{W}$$

A parte seguinte da análise (pontos 3.d - 3.h abaixo) é executada em uma das três circunstâncias que seguem:

- quando o passo precede uma queda do percentual de casos corretamente classificados,
- quando o percentual de casos corretamente classificados é igual a 100,
- quando o passo é o último.

d) ***Alocação e distâncias de casos na amostra básica.*** As distâncias em relação a cada grupo são calculadas como descrito no ponto 3.a acima; as variáveis usadas no cálculo são aquelas retidas no passo. A designação de casos para os grupos é feita como descrito no ponto 3.a acima.

e) ***Análise fatorial discriminante.*** A matriz $\mathcal{T}_q^{-1} \, \mathcal{B}_q$ descrita em 3.c acima é analisada. Os dois primeiros vetores próprios correspondentes aos dois maiores valores próprios dessa matriz serão os dois eixos fatoriais discriminantes. A potência discriminante dos fatores é medida pelos valores próprios correspondentes. Como o programa fornece a potência discriminante para os três primeiros fatores, a soma dos valores próprios permite estimar o nível de valores próprios remanescentes, i.e. aqueles que não são impressos.

f) ***Valores de fatores discriminantes para todos os casos e médias de grupos.***

Para um CASO, o valor do fator discriminante é calculado como o produto escalar do vetor do caso contendo variáveis retidas no passo pelo valor próprio correspondendo ao fator. Note que esses valores não são impressos, mas eles são usados na representação gráfica dos casos no espaço dos dois primeiros fatores.

Para uma MÉDIA DE GRUPO, o valor do fator discriminante é calculado da mesma maneira, mudando-se o vetor de caso para o vetor de média de grupo.

g) ***Alocação e distâncias dos casos na amostra de teste.*** As distâncias em relação a cada grupo são calculadas do mesmo jeito, e a designação de casos para os grupos é feita seguindo as mesmas regras da amostra básica (ver 3.d acima).

h) ***Alocação e distâncias dos casos na amostra anônima.*** As distâncias em relação a cada grupo são calculadas do mesmo jeito e a designação de casos para os grupos é feita seguindo-se as mesmas regras da amostra básica (ver 3.d acima).

44.4 Referências

Romeder, J.M., *Méthodes et programmes d'analyse discriminante*, Dunod, Paris, 1973.

Capítulo 45

Funções de Distribuição e de Lorenz

Notação

p_i = valor do i-ésimo ponto de quebra
i = subscrito para ponto de quebra
s = número de subintervalos
N = número total de casos.

45.1 Fórmula para Pontos de Quebra

O número de pontos de quebra é um a menos do que o número de subintervalos requisitados, e.g. medianas implicam dois subintervalos e um ponto de quebra.

$$p_i = \mathcal{V}(\alpha) + \beta\left[\mathcal{V}(\alpha+1) - \mathcal{V}(\alpha)\right]$$

onde $\mathcal{V}$ é vetor de dados ordenados, e.g. $\mathcal{V}(3)$ é o terceiro item no vetor,

$$\alpha = \text{entier}\left[\frac{i(N+1)}{s}\right]$$

$$\beta = \frac{i(N+1)}{s} - \alpha$$

e entier(x) é o maior interiro que não excede x.

45.2 Pontos de Quebra de Função de Distribuição

Há quatro situações possíveis:

- Se um ponto de quebra se localiza exatamente em um valor e o valor não está empatado com nenhum outro valor, então o valor será o ponto de quebra.
- Se um ponto de quebra se localiza exatamente entre dois valores e os dois valores não são os mesmos, então o ponto de quebra é determinado usando-se interpolação linear ordinária.
- Se um ponto de quebra se localiza exatamente em um valor e o valor está empatado com um ou mais valores, então o procedimento involve o cômputo de novos pontos médios. Seja k o valor, m a freqüência com a qual ele ocorre e d a distânica mínima entre itens no vetor $\mathcal{V}$. O intervalo $k \pm \min(d,1)/2$ é dividido em m partes e pontos médios são computados para esses novos intervalos. O ponto de quebra é, então, o ponto médio apropriado.
- Se um ponto de quebra se localiza entre dois valores que são idênticos, o procedimento envolve tanto o cálculo de novos pontos médios quanto interpolação linear. Seja k o valor, m a freqüência com a

qual ele ocorre e d a distânica mínima entre itens no vetor $\mathcal{V}$. O intervalo $k \pm \min(d, 1)/2$ é dividido em m partes e pontos médios são computados para esses novos intervalos. Então, interpolação linear é executada entre os dois novos pontos apropriados.

45.3 Pontos de Quebra da Função de Lorenz

Para determinar os pontos de quebra da função de Lorenz, o vetor de dados ordenado é acumulado e, a cada passo, o total acumulado é dividido pelo total geral. Então os pontos de quebra são encontrados do mesmo jeito que descrito acima.

45.4 Curva de Lorenz

A função de Lorenz plotada contra a proporção da população ordenada fornece a curva de Lorenz, que está sempre contida no triângulo inferior do quadrado unitário. O programa QUANTILE usa dez subintervalos para a curva de Lorenz.

Note que os valores da função de Lorenz são chamados de "fração de riqueza" na impressão.

45.5 O coeficiente de Gini

O coeficiente de Gini representa duas vezes a área entre a função de Lorenz e a diagonal impressa no quadrado unitário. Ele assume valores entre 0 e 1. Zero (0) indica "igualdade perfeita" - todos os valores de dados são iguais. One (1) indica "desigualdade perfeita" - há um valor de dados não-zero.

O programa usa uma aproximação:

$$\text{Coeficiente de Gini} = 1 - \frac{1}{s} - \frac{2}{s}\sum_{i=1}^{s-1} l_i$$

onde l_i é o $i^{'esimo}$ ponto de quebra da função Lorenz.

Essa aproximação se torna mais exata à medida que o número de pontos de quebra é aumentado; é recomendado que pelo menos dez sejam utilizados.

45.6 Estatística D de Kolmogorov-Smirnov

O teste de Kolmogorov-Smirnov tem a ver com a concordância de duas distribuições cumulativas. Se duas distribuições cumulativas amostrais estão muito longe em qualquer ponto, isso sugere que as amostras vêm de populações diferentes. O teste é focado na maior diferença entre duas distribuições.

Sejam $\mathcal{V}_1$ e $\mathcal{V}_2$ vetores de dados ordenados para a primeira e segunda variável, respectivamente, e $\mathcal{X}$ o vetor de códigos que aparece em ambas as distribuições. O programa cria as duas funções escada cumulativas $F_1(x)$ e $F_2(x)$, respectivamente. Então ele procura pela diferença absoluta máxima entre as distribuições,

$$D = \max(|F_1(x) - F_2(x)|)$$

e imprime:

x : o valor onde a primeira diferença absoluta máxima ocorre
f_1 : o valor de F_1 associado com x
f_2 : o valor de F_2 associado com x.

Se os N's para $\mathcal{V}_1$ e $\mathcal{V}_2$ são iguais e menores do que 40, o programa imprime a estatística K igual à diferença em freqüências associadas com a diferença máxima. Uma tabela de valores críticos da estastística K, denotado por K_D, pode ser consultada para determinar a significância da diferença observada.

Se os N's para $\mathcal{V}_1$ e $\mathcal{V}_2$ não são iguais ou maiores que 40, o programa imprime as seguintes estatísticas:

$$\text{desvio não-ajustado} = D = |f_1 - f_2|$$

$$\text{desvio ajustado} = D \sqrt{\frac{N_1\, N_2}{N_1 + N_2}}$$

onde N_1 e N_2 são iguais ao número de casos em $\mathcal{V}_1$ e $\mathcal{V}_2$, respectivamente.

$$\text{Aproximação chi-quadrado} = 4D^2\, \frac{N_1\, N_2}{N_1 + N_2}$$

Note: A significância do desvio direcional máximo pode ser encontrada ao se considerar o valor dessa chi-quadrado como a distribuição de uma chi-quadrado com dois graus de liberdade.

45.7 Nota a Respeito de Pesos

Para pontos de quebra de função de distribuição, pontos de quebra de função de Lorenz e coeficientes de Gini, os dados podem ser ponderados por um inteiro. Se o peso for especificado, cada caso será implicitamente contado como "w" casos, onde "w" é o valor do peso para o caso. O teste de Kolmogorov-Smirnov é sempre executado com dados não-ponderados.

Capítulo 46

Análise Fatorial

Notação

x	=	valores das variáveis
i	=	subscrito para caso
j, j'	=	subscritos para variáveis
α	=	subscrito para fator
m	=	número de fatores determinados /desejados
$I1$	=	número de casos ativos
$J1$	=	número de variáveis ativas
w	=	valor do peso
W	=	soma total de pesos para casos ativos.

46.1 Estatísticas Univariadas

Essas estatísticas univariadas são calculadas para todas as variáveis usadas na análise, i.e. variáveis ativas e passivas, se houverem. Note que as variáveis são renumeradas de 1 (coluna RNK). Somente casos ativos entram nos cálculos.

a) ***Média.***

$$\overline{x}_j = \frac{\sum_{i=1}^{I1} w_i \, x_{ij}}{W}$$

b) ***Variância (estimada).***

$$\widehat{s_j}^2 = \left(\frac{N}{N-1}\right)\left[\frac{W\sum_{i=1}^{I1} w_i \, x_{ij}^2 - \left(\sum_{i=1}^{I1} w_i \, x_{ij}\right)^2}{W^2}\right]$$

c) ***Desvio-padrão (estimado).***

$$\widehat{s_j} = \sqrt{\widehat{s_j}^2}$$

d) ***Coeficiente de variação (C. Var.).***

$$C_j = \frac{\widehat{s_j}}{\overline{x}_j}$$

e) ***Total (soma para x_j).***

$$Total_j = \sum_{i=1}^{I1} w_i \, x_{ij}$$

f) ***Assimetria.***

$$g1_j = \frac{m3_j}{\widehat{s}_j^2 \sqrt{\widehat{s}_j^2}} \quad \text{onde} \quad m3_j = \frac{\sum_{i=1}^{I1} w_i \, (x_{ij} - \overline{x}_j)^3}{W}$$

g) ***Curtose.***

$$g2_j = \frac{m4_j}{(\widehat{s}_j^2)^2} - 3 \quad \text{onde} \quad m4_j = \frac{\sum_{i=1}^{I1} w_i \, (x_{ij} - \overline{x}_j)^4}{W}$$

h) ***N Ponderado.*** Número de casos ativos, se o peso não for especificado, ou número ponderado de casos ativos (soma dos pesos).

46.2 Dados de Entrada

Os dados são impressos para os casos ativos e passivos.

A primeira coluna da tabela contém os valores da variable de ID de caso (até 4 dígitos). A segunda coluna (Coef) contém o valor do peso designado para cada caso (w_i). A terceira coluna (PI) é igual a soma ponderada dos valores das variáveis ativas, para cada caso (totais de linha ponderados).

$$P_{i\cdot} = \sum_{j=1}^{J1} w_i \, x_{ij}$$

A primeira linha contém os primeiros quatro caracteres de cada nome de variável. A segunda linha (PJ) é igual a soma ponderada dos valores dos casos ativos, para cada variável (totais de coluna ponderados).

$$P_{\cdot j} = \sum_{i=1}^{I1} w_i \, x_{ij}$$

Note que o valor do "Coef" no início dessa linha é igual ao número ponderado de casos ativos, e o valor de "PI" é igual ao Total (P) das variáveis ativas para os casos ativos.

$$P = \sum_{i=1}^{I1} P_{i\cdot} = \sum_{j=1}^{J1} P_{\cdot j} = \sum_{i=1}^{I1} \sum_{j=1}^{J1} w_i \, x_{ij}$$

O resto da tabela de dados de entrada contém os valores (com uma casa decimal) das variáveis ativas e passivas.

46.3 Matrizes Núcleo (Matrizes de Relações)

Para cada tipo de análise, a matriz núcleo é calculada e impressa. Essa é uma matriz de relações entre as variáveis. Note que, para os resultados, os valores da matriz são multiplicados por um fator cujo valor é impresso próximo do título da matriz. Esse fator é igualado a zero quando alguns valores na matriz excederem 5 caracteres (pode ser o caso de produtos escalares ou matrizes de covariâncias).

Para a ANÁLISE DE CORRESPONDÊNCIAS, os elementos $C_{jj'}$ da matriz núcleo são calculados da seguinte maneira:

$$C_{jj'} = \frac{1}{\sqrt{P_{\cdot j}}\sqrt{P_{\cdot j'}}} \sum_{i=1}^{I1} \frac{(w_i\, x_{ij})\,(w_i\, x_{ij'})}{P_{i\cdot}}$$

Para a ANÁLISE DE PRODUTOS ESCALARES, os elementos $SP_{jj'}$ da matriz núcleo são calculados da seguinte maneira:

$$SP_{jj'} = \sum_{i=1}^{I1} w_i\, x_{ij}\, x_{ij'}$$

Para a ANÁLISE DE PRODUTOS ESCALARES NORMADOS, os elementos $NSP_{jj'}$ da matriz núcleo são calculados da seguinte maneira:

$$NSP_{jj'} = \frac{\displaystyle\sum_{i=1}^{I1} w_i\, x_{ij}\, x_{ij'}}{\sqrt{\Big(\displaystyle\sum_{i=1}^{I1} w_i\, x_{ij}^2\Big)\Big(\displaystyle\sum_{i=1}^{I1} w_i\, x_{ij'}^2\Big)}}$$

Para a ANÁLISE DE COVARIÂNCIAS, os elementos $COV_{jj'}$ da matriz núcleo são calculados da seguinte maneira:

$$COV_{jj'} = \frac{\displaystyle\sum_{i=1}^{I1} w_i\,(x_{ij} - \overline{x}_j)\,(x_{ij'} - \overline{x}_{j'})}{W}$$

Para a ANÁLISE DE CORRELAÇÕES, os elementos $COR_{jj'}$ da matriz núcleo são calculados da seguinte maneira:

$$COR_{jj'} = \frac{\displaystyle\sum_{i=1}^{I1} w_i\,(x_{ij} - \overline{x}_j)\,(x_{ij'} - \overline{x}_{j'})}{\sqrt{\displaystyle\sum_{i=1}^{I1} w_i\,(x_{ij} - \overline{x}_j)^2 \sum_{i=1}^{I1} w_i\,(x_{ij'} - \overline{x}_{j'})^2}}$$

46.4 Traço

Traço da matriz núcleo é calculado como a soma dos seus elementos da diagonal. O traço é igual ao total dos valores próprios (inércia total). Note que, para a análise de correlações e análise de produtos escalares normados, a inércia total é igual ao número de variáveis ativas.

$$\text{Traço} = \sum_{\alpha=1}^{J1} \lambda_\alpha$$

46.5 Valores e Vetores Próprios

Os valores próprios e vetores próprios são impressos para os fatores retidos. Eles possuem o mesmo significado para cada tipo de análise, mas são de pouco interesse para o usuário.

Para a análise de correspondências, o programa imprime um valor próprio e vetor próprio a mais do que o número de fatores determinados/desejados. O fator para o valor próprio trivial (sendo sempre igual a 1) é impresso como o primeiro e é negligenciado mais tarde. Os fatores remanescentes são renumerados (começando de 1) nas tabelas de variáveis/casos ativos/passivos.

46.6 Tabela de Valores Próprios

A tabela contém todos os valores próprios, denotados aqui por λ_α, calculados pelo programa. Note que em análise de correspondências, o primeiro, valor próprio trivial (sendo sempre 1) é impresso somente sobre a tabela e seu valor é subtraído do traço ao calcular-se o percentual no ponto 6.d abaixo.

a) ***NO.*** Número seqüencial do valor próprio, α, em ordem crescente.

b) ***ITER.*** Número de iterações usadas na computação de vetores próprios correspondentes. Valor zero significa que o vetor próprio correspondente foi obtido ao mesmo tempo que o anterior (de baixo para cima).

c) ***Val próprio.*** Essa coluna dá a seqüência de valores próprios, lambdas, cada um correspondendo ao fator α.

d) ***Percent.*** Contribuição do fator para a inércia total (em termos de percentuais).

$$\tau_\alpha = \frac{\lambda_\alpha}{\text{Traço}} \times 100$$

e) ***Cumul (percentuais cumulativos).*** Contribuição dos fatores 1 até α para a inércia total (em termos de percentagens).

$$\text{Cumul}_\alpha = \tau_1 + \tau_2 + \cdots + \tau_\alpha$$

f) ***Histograma de valores próprios.*** Cada valor próprio é representado por uma linha de asteriscos cujo o número é proporcional ao valor próprio. O primeiro valor próprio no histograma é sempre representado por 60 asteriscos. O histograma permite uma análise visual da diminuição relativa dos valores próprios para fatores subseqüentes.

46.7 Tabela de Fatores de Variáveis Ativas

A tabela contém as ordenadas das variáveis ativas no espaço fatorial, seus cossenos ao quadrado com cada fator e suas contribuições para cada fator. Além disso, ela contém a qualidade dessas variáveis, seus pesos e suas inércias.

a) ***JPR.*** Número da variável para as variáveis ativas.

b) ***QLT.*** A qualidade de representação da variável no espaço de m fatores é medida, para TODOS OS TIPOS DE ANÁLISE, pela soma dos cossenos ao quadrado (ver 7.f abaixo). Valores próximos de 1 indicam um nível maior de representação da variável por meio dos fatores.

$$QLT_j = \sum_{\alpha=1}^{m} COS2_{\alpha\, j}$$

c) ***PESO.*** Valor do peso da variável. Para TODOS OS TIPOS DE ANÁLISE, ele é calculado como o quociente entre o total da variável e o Total geral (ver seção 2 acima), multiplicado por 1000.

$$f_{\cdot j} = \frac{P_{\cdot j}}{P} \times 1000$$

Note que o peso (PESO) impresso na última linha da tabela é igual a:

- o Total geral, para a análise de correspondência,
- o número de casos ponderados, para outros tipos de análise.

d) ***INR.*** Inércia correspondendo à variável. Ela indica a parte da inércia total relacionada à variável no espaço de fatores.

Para a ANÁLISE DE CORRESPONDÊNCIAS, ela é calculada como o quociente entre a inércia da variável e a inércia total, multiplicado por 1000. Note que a inércia da variável depende do peso da variável e que o valor do Traço usado aqui não inclui os valores próprios triviais.

$$INR_j = \frac{f_{\cdot j} \sum_{\alpha=1}^{J1-1} F_{\alpha\, j}^2}{\text{Traço}} \times 1000$$

onde $F_{\alpha\, j}$ é a ordenada da variável j correspondendo ao fator α (ver 7.e abaixo).

Para a ANÁLISE DE PRODUTOS ESCALARES e ANÁLISE DE COVARIÂNCIAS, a inércia da variável não depende do peso da variável.

$$INR_j = \frac{\sum_{\alpha=1}^{J1} F_{\alpha\, j}^2}{\text{Traço}} \times 1000$$

Para a ANÁLISE DE PRODUTOS ESCALARES NORMADOS e a ANÁLISE DE CORRELAÇÕES, a inércia da variável depende somente do número de variáveis ativas.

$$INR_j = \frac{1}{J1} \times 1000$$

Note que a inércia (INR) impressa na última linha da tabela é igual a 1000.

As três colunas seguintes são repetidas para cada fator.

e) $\alpha\#F$**.** A ordenada da variável no espaço de fator, denotada aqui por $F_{\alpha\, j}$.

f) ***COS2.*** Cosseno ao quadrado do ângulo entre a variável e o fator. É uma medida de "distância" entre a variável e o fator. Valores próximos de 1 indicam distâncias pequenas em relação a um fator.

Para a ANÁLISE DE CORRESPONDÊNCIAS, calcula-se da seguinte maneira:

$$COS2_{\alpha\, j} = \frac{F_{\alpha\, j}^2}{\sum_{\alpha=1}^{J1-1} F_{\alpha\, j}^2} \times 1000$$

Para a ANÁLISE DE PRODUTOS ESCALARES e para ANÁLISE DE COVARIÂNCIAS,

$$COS2_{\alpha\, j} = \frac{F_{\alpha\, j}^2}{\sum_{\alpha=1}^{J1} F_{\alpha\, j}^2} \times 1000$$

Para a ANÁLISE DE PRODUTOS ESCALARES NORMADOS e para ANÁLISE DE CORRELAÇÕES,

$$COS2_{\alpha\, j} = F_{\alpha\, j}^2 \times 1000$$

g) ***CPF.*** Contribuição da variável para o fator.

Para a ANÁLISE DE CORRESPONDÊNCIAS,

$$CPF_{\alpha\, j} = \frac{f_{\cdot j}\, F_{\alpha\, j}^2}{\lambda_\alpha} \times 1000$$

Para TODOS OS OUTROS TIPOS DE ANÁLISE,

$$CPF_{\alpha\, j} = \frac{F_{\alpha\, j}^2}{\lambda_\alpha} \times 1000$$

Note que a contribuição (CPF) impressa na última linha da tabela é igual a 1000.

46.8 Tabela de Fatores de Variáveis Passivas

A tabela contém a mesma informação daquela descrita no ponto 7. acima, exceto para as variáveis passivas.

a) ***JSUP.*** Número da variável para as variáveis passivas.

b) ***QLT.*** Qualidade da representação da variável no espaço dos m fatores (ver 7.b acima).

c) ***PESO.*** Valor do peso da variável (ver 7.c acima).

d) ***INR.*** Inércia correspondendo a variável. Note que as variáveis passivas não contribuem para a inércia total. Portanto, a inércia aqui indica se a variável pode exercer algum papel na análise, caso ela seja usada com variável principal. Ela é calculada do mesmo jeito que para as variáveis ativas nas análises respectivas (ver 7.d acima).

A inércia (INR) impressa na última linha da tabela é igual ao total INR ao longo de todas as variáveis passivas.

As três colunas seguintes são repetidas para cada fator.

e) $\alpha\#F$**.** A ordenada da variável no espaço de fator, denotada aqui por $F_{\alpha\, j}$.

f) ***COS2.*** Cosseno quadrado do ângulo entre a variável e o fator. É calculado da mesma maneira como para as variáveis ativas nas análises respectivas (ver 7.f acima).

g) ***CPF.*** Contribuição da variável para o fator. Note as variáveis passivas não participam na construção do espaço de fator. Portanto, a contribuição somente indica se a variável poderia ter algum papel na análise, caso ela fosse usada como a variável principal. CPF é calculado do mesmo jeito que as variáveis ativas nas análises respectivas (ver 7.g acima).

A contribuição (CPF) impressa na última linha da tabela é igual ao CPF total ao longo de todas as variáveis passivas.

46.9 Tabela de Fatores de Casos Ativos

A tabela contém as ordenadas dos casos ativos no espaço fatorial, seus cossenos ao quadrado com cada fator e suas contribuições para cada fator. Além disso, ela contém a qualidade de representação desses casos, seus pesos e suas inércias.

a) ***IPR.*** Valor de ID de caso para os casos ativos.

b) ***QLT.*** Qualidade de representação do caso no espaço de m fatores é medida, para TODOS OS TIPOS DE ANÁLISE, pela soma dos cossenos ao quadrado (ver 9.f abaixo). Valores próximos de 1 indicam o nível mais alto de representação do caso pelos fatores.

$$QLT_i = \sum_{\alpha=1}^{m} COS2_{\alpha\, i}$$

c) ***PESO.*** Valor do peso para o caso.

Para a ANÁLISE DE CORRESPONDÊNCIAS, ele é calculado como o quociente entre a soma (ponderada) das variáveis ativas para esse caso e o Total geral (ver seção 2 acima), multiplicado por 1000.

$$f_{i\cdot} = \frac{P_{i\cdot}}{P} \times 1000$$

Note que o peso (PESO) impresso na última linha da tabela é igual ao Total geral.

Para TODOS OS OUTROS TIPOS DE ANÁLISE,

$$f_{i\cdot} = \frac{w_i}{P} \times 1000$$

Note que o peso (PESO) impresso na última linha da tabela é igual ao número de casos ponderado.

d) ***INR.*** Inércia correspondendo ao caso. Isso indica a parte da inércia total relacionada com o caso no espaço de fatores.

Para a ANÁLISE DE CORESPONDÊNCIAS, ela é calculada como o quociente entre a inércia do caso e a inércia total, multiplicado por 1000. Note que a inércia do caso depende do peso do caso e que o valor do Traço usado aqui não inclui o valor próprio trivial.

$$INR_i = \frac{f_{i\cdot} \sum_{\alpha=1}^{J1-1} F_{\alpha\, i}^2}{\text{Traço}} \times 1000$$

Para TODOS OS OUTROS TIPOS DE ANÁLISE,

$$INR_i = \left(\frac{w_i}{W \times \text{Traço}} \sum_{j=1}^{J1} z_{ij}^2 \right) \times 1000$$

onde

$$z_{ij} = \begin{cases} \dfrac{x_{ij}}{x_{ij}} & \text{para análise de produtos escalares} \\ \dfrac{x_{ij}}{\sqrt{\left(\sum_{i=1}^{I1} w_i\, x_{ij}^2\right) / W}} & \text{para análise de produtos escalares normados} \\ x_{ij} - \overline{x}_j & \text{para análise de covariâncias} \\ \dfrac{x_{ij} - \overline{x}_j}{s_j} & \text{para análise de correlações} \end{cases}$$

e s_j é o desvio-padrão amostral da variável j.

Note que a inércia (INR) impressa na última linha da tabela é igual a 1000.

As três colunas seguintes são repetidas para cada fator.

e) $\alpha\#F$**.** A ordenada do caso no espaço de fatores, denotado aqui por $F_{\alpha\, i}$.

f) ***COS2.*** Cosseno ao quadrado do ângulo entre o caso e o fator. É uma medida da "distância" entre o caso e o fator. Valores próximos de 1 indicam distâncias menores em relação ao fator.

Para a ANÁLISE DE CORRESPONDÊNCIAS, ele é calculado da seguinte maneira:

$$COS2_{\alpha\, i} = \frac{F_{\alpha\, i}^2}{\sum_{\alpha=1}^{J1-1} F_{\alpha\, i}^2} \times 1000$$

Para TODOS OS OUTROS TIPOS DE ANÁLISE,

$$COS2_{\alpha\, i} = \frac{F_{\alpha\, i}^2}{\sum_{\alpha=1}^{J1} F_{\alpha\, i}^2} \times 1000$$

g) ***CPF.*** Contribuição do caso para o fator.

Para a ANÁLISE DE CORRESPONDÊNCIAS,

$$CPF_{\alpha\, i} = \frac{f_{i\cdot}\, F_{\alpha\, i}^2}{\lambda_\alpha} \times 1000$$

Para TODOS OS OUTROS TIPOS DE ANÁLISE,

$$CPF_{\alpha\, i} = \frac{w_i\, F_{\alpha\, i}^2}{W\, \lambda_\alpha} \times 1000$$

Note que a contribuição (CPF) impressa na última linha da tabela é igual a 1000.

46.10 Tabela de Fatores de Casos Passivos

A tabela contém a mesma informação que aquela descrita no ponto 9. acima, com exceção dos casos passivos.

a) ***ISUP.*** Valor de ID de caso para os casos passivos.

b) ***QLT.*** Qualidade de representação do caso no espaço de m fatores (ver 9.b acima).

c) ***PESO.*** Valor do peso do caso (ver 9.c acima).

d) ***INR.*** Inércia correspondendo ao caso. Note que os casos passivos não contribuem para a inércia total. Portanto, a inércia aqui indica se o caso poderia ter algum papel na análise, caso ele fosse usado como o principal. Ela é calculada do mesmo jeito que para os casos ativos nas análises respectivas (ver 9.d acima).

A inércia (INR) impressa na última linha da tabela é igual ao INR total ao longo de todos os casos passivos.

As três colunas seguintes são repetidas para cada fator.

e) $\alpha\#F$***.*** A ordenada para o caso no espaço de fator, denotada aqui por $F_{\alpha\, i}$.

f) ***COS2.*** Cosseno ao quadrado do ângulo entre o caso e o fator. É calculado do mesmo jeito dos casos ativos nas análises respectivas (see 9.f above).

g) ***CPF.*** Contribuição do caso para o fator. Note que os casos passivos não participam na construção do espaço de fator. Portanto, a contribuição indica somente se o caso poderia ter algum papel na análise, caso ele fosse usado como o principal. CPF é calculada da mesma maneira que para os casos ativos nas análises respectivas (ver 9.g acima).

A contribuição (CPF) impressa na última linha da tabela é igual ao CPF total ao longo de todas os casos passivos.

46.11 Fatores Rotacionados

Aplicado apenas para análise de correlação. Os fatores de "variável" podem ser rotacionados ao terminar-se a análise fatorial. O procedimento Varimax usado aqui é o mesmo que o usado no programa CONFIG. Note que os fatores de "variável" para as variáveis ativas podem ser tratados como uma configuração de $J1$ objetos em espaço dimensional α.

46.12 Referências

Benzécri, J.-P. and F., *Pratique de l'analyse de données*, tome 1: *Analyse des correspondances*, exposé élémentaire, Dunod, Paris, 1984.

Iagolnitzer, E.R., Présentation des programmes MLIFxx d'analyses factorielles en composantes principales, *Informatique et sciences humaines*, 26, 1975.

Capítulo 47

Regressão Linear

Notação

y	=	valor da variável dependente
x	=	valor de uma variável independente (explanatória)
i, j, l, m	=	subscritos para variáveis
p	=	número de preditores
k	=	subscrito par caso
N	=	número total de casos
w	=	valor do peso multiplicado por $\frac{N}{W}$
W	=	soma total dos pesos.

47.1 Estatísticas Univariadas

Essas estatísticas ponderadas são calculadas para todas as variáveis usadas na análise, i.e., variáveis independentes e variável dependente.

a) ***Média.***

$$\overline{x}_i = \frac{\sum_k w_k \, x_{ik}}{N}$$

b) ***Desvio-padrão (estimado).***

$$\widehat{s}_i = \sqrt{\frac{N \sum_k \left(w_k \, x_{ik}\right)^2 \; - \; \left(\sum_k w_k \, x_{ik}\right)^2}{N(N-1)}}$$

c) ***Coeficiente de variação (C.var.).***

$$C_i = \frac{100 \, \widehat{s}_i}{\overline{x}_i}$$

47.2 Matriz de Soma Total de Quadrados e Produtos Cruzados

É calculado para todas as variáveis usadas na análise como segue:

$$\text{t.s.s.c.p.}_{\,ij} = \sum_k w_k \, x_{ik} \, x_{jk}$$

47.3 Matriz de Soma de Quadrados Residuais e Produtos Cruzados

Essa matriz, às vezes chamada matriz dos quadrados e produtos cruzados dos escores dos desvios, é calculada para todas as variáveis usadas na análise, como segue:

$$\text{r.s.s.c.p.}_{ij} = \sum_k w_k\, x_{ik}\, x_{jk} \;-\; \frac{\left(\sum_k w_k\, x_{ik}\right)\left(\sum_k w_k\, x_{jk}\right)}{N}$$

47.4 Matriz de Correlação Total

Os elementos dessa matriz são calculados diretamente da matriz de soma de quadrados residuais e produtos cruzados. Note que se essa fórmula for escrita em detalhe e se numerador e denominador forem ambos multiplicados por N, tem-se uma fórmula convencional para o r de Pearson.

$$r_{ij} = \frac{\text{r.s.s.c.p.}_{ij}}{\sqrt{\text{r.s.s.c.p.}_{ii}}\,\sqrt{\text{r.s.s.c.p.}_{jj}}}$$

47.5 Matriz de Correlação Parcial

O $ij^{'esimo}$ elemento dessa matriz é o coeficiente de correlação parcial entre a variável i e a variável j, mantendo constantes as variáveis especificadas. Correlações parciais descrevem o grau de correlação que existiria entre duas variáveis dado que a variação em outras variáveis está controlada. Elas também descrevem a correlação entre variáveis independentes (explanatória) que seriam selecionadas em uma regressão stepwise.

a) ***Correlação entre*** x_i ***e*** x_j ***mantendo constante*** x_l (coeficientes de correlação parcial de primeira-ordem).

$$r_{ij\cdot l} = \frac{r_{ij} \;-\; r_{il}\, r_{jl}}{\sqrt{1 - r_{il}^2}\,\sqrt{1 - r_{jl}^2}}$$

onde r_{ij}, r_{il}, r_{jl} são coeficientes de ordem-zero (coeficientes r de Pearson).

b) ***Correlação entre*** x_i ***e*** x_j ***mantendo constante*** x_l ***e*** x_m (coeficientes de correlação parcial de segunda-ordem).

$$r_{ij\cdot lm} = \frac{r_{ij\cdot l} \;-\; r_{im\cdot l}\, r_{jm\cdot l}}{\sqrt{1 - r_{im\cdot l}^2}\,\sqrt{1 - r_{jm\cdot l}^2}}$$

onde $r_{ij\cdot l}$, $r_{im\cdot l}$, $r_{jm\cdot l}$ são coeficientes de primeira-ordem.

Note: O programa computa as correlações trabalhando, passo a passo, a partir de coeficientes de ordem-zero até coeficientes de primeira-ordem, segunda-ordem, etc.

47.6 Matriz Inversa

Para uma regressão padrão, essa é a inversa da matriz de correlação das variáveis independentes (explanatórias) e a variável dependente. Para uma regressão stepwise, essa é a inversa da matriz de correlação das variáveis independentes na equação final. O programa usa o método de eliminação Gaussiana para a inversão.

47.7 Estatísticas de Resumo de Análise

a) ***Erro padrão de estimação.*** Esse é o desvio-padrão dos resíduos.

$$\text{Erro padrão de estimação} = \sqrt{\frac{\sum_k (y_k - \widehat{y}_k)^2}{\text{df}}}$$

onde

$$\begin{aligned} \widehat{y}_k &= \text{o valor previsto da variável dependente para o } k'^{esimo} \text{ caso} \\ \text{df} &= \text{graus de liberdade dos resíduos (ver 7.f abaixo).} \end{aligned}$$

b) ***F-ratio para a regressão.*** Essa é a estatística F para determinar a significância estatística do modelo sob consideração. Os graus de liberdade são p e $N - p - 1$.

$$F = \frac{R^2 \,\text{df}}{p\,(1 - R^2)}$$

onde R^2 é a fração da variância explicada (ver 7.d abaixo).

c) ***Coeficiente de correlação múltipla.*** Essa é a correlação entre a variável dependente e o escore preditor. Ele indica a intensidade da associação entre o critério e a função linear dos preditores, e é similar a um coeficiente de correlação de Pearson simples, exceto pelo fato de ser sempre positivo.

$$R = \sqrt{R^2}$$

R não é impresso se o termo constante for restrito a zero.

d) ***Fração da variância explicada.*** R^2 pode ser interpretado como a proporção da variância na variável dependente explicada pelos preditores. Às vezes chamado de coeficiente de determinação, é uma medida da efetividade geral da regressão linear. Quanto maior ele for, melhor a equação ajustada explica a variação nos dados.

$$R^2 = 1 - \frac{\sum_k (y_k - \widehat{y}_k)^2}{\sum_k (y_k - \overline{y})^2}$$

onde

$$\begin{aligned} \widehat{y}_k &= \text{o valor predito da variável dependente para o } k'^{esimo} \text{ caso} \\ \overline{y} &= \text{a média da variável dependente.} \end{aligned}$$

Como R, R^2 não é impresso se o termo constante for restringido a zero.

e) ***Determinante da matriz de correlação.*** Esse é o determinante da matriz de correlação dos preditores. Ele representa, através de um único número, a variância generalizada em um conjunto de variáveis, e varia de 0 a 1. Determinantes próximos de zero indicam que algumas ou todas as variáveis explanatórias são altamente correlacionadas. Um determinante de zero indica uma matriz singular, o que significa que pelo menos um dos preditores é uma função linear de um ou outros mais.

f) ***Graus de liberdade dos resíduos.***

Se a constante não é restrita a ser zero,

$$\text{df} = N - p - 1$$

Se a constante é restrita a ser zero,

$$\text{df} = N - p$$

g) ***Termo constante.***

$$A = \overline{y} - \sum_i B_i \, \overline{x}_i$$

onde

$$\begin{aligned} \overline{y} &= \text{a média da variável dependente (ver 1.a acima)} \\ \overline{x}_i &= \text{a média da variável preditora } i \text{ (ver 1.a acima)} \\ B_i &= \text{o coeficiente B para a variável preditora } i \text{ (ver 8.a abaixo).} \end{aligned}$$

47.8 Estatísticas de Análise para Preditores

a) ***B.*** Essas são coeficientes de regressão parciais não-padronizados que são apropriados (ao invés dos betas) para serem utilizados em uma equação com a finalidade de prever escores brutos. Eles são sensíveis à escala de medida da variável preditora e à variância da variável preditora.

$$B_i = \beta_i \, \frac{\widehat{s}_y}{\widehat{s}_i}$$

onde

$$\begin{aligned} \beta_i &= \text{o peso beta para o preditor } i \text{ (ver 8.c abaixo)} \\ \widehat{s}_y &= \text{o desvio-padrão da variável dependente (ver 1.b acima)} \\ \widehat{s}_i &= \text{o desvio-padrão da variável preditora } i \text{ (ver 1.b acima).} \end{aligned}$$

b) ***Sigma B.*** Esse é o desvio-padrão de B, uma medida da confiabialidade do coeficiente.

$$\text{Sigma } B_i = (\text{erro padrão de estimação}) \, \sqrt{\frac{c_{ii}}{\text{r.s.s.c.p.}_{ii}}}$$

onde c_{ii} é o $i^{'esimo}$ elemento da diagonal da inversa da matriz de correlação dos preditores na equação de regressão (ver a seção 6 acima).

c) ***Beta.*** Esses coeficientes de regressão são também chamados de "coeficientes de regressão parcial padronizados" ou "coeficientes B padronizados". Eles são independentes da escala de medida. As magnitudes dos quadrados dos betas indicam as contribuições relativas das variáveis para a predição.

$$\beta_i = \mathcal{R}_{11}^{-1} \, \mathcal{R}_{yi}$$

onde

$$\begin{aligned} \mathcal{R}_{11} &= \text{matriz de correlação dos preditores na equação} \\ \mathcal{R}_{yi} &= \text{vetor coluna das correlações da variável dependente e preditores} \\ &\quad \text{indicados pelo preditor } i. \end{aligned}$$

d) ***Sigma Beta.*** Esse é o erro padrão do coeficiente beta, uma medida da confiabilidade do coeficiente.

$$\text{Sigma}\, \beta_i = \text{sigma}\, B_i \, \frac{\widehat{s}_i}{\widehat{s}_y}$$

e) ***R quadrado parcial.*** Essas são correlações parciais, ao quadrado, entre os preditores i e a variável dependente, y, com a influência das outras variáveis na equação de regressão eliminada. O coeficiente de correlação parcial ao quadrado é uma medida do grau que aquela parte da variação na variável dependente, não explicada por outros preditores, pode ser explicada pelo preditor i.

$$r^2_{yi \cdot \, jl\ldots} = \frac{R^2_{y \cdot \, ijl\ldots} - R^2_{y \cdot \, jl\ldots}}{1 - R^2_{y \cdot \, jl\ldots}}$$

onde

$$\begin{aligned} R^2_{y\cdot\, ijl\ldots} &= R \text{ quadrado múltiplo com preditor } i \\ R^2_{y\cdot\, jl\ldots} &= R \text{ quadrado múltiplo sem preditor } i. \end{aligned}$$

f) ***R quadrado marginal.*** Isso é o aumento na variância explicada ao adicionar-se o preditor i aos outros preditores na equação de regressão.

$$r_i^2\, marginal = R^2_{y\cdot\, ijl\ldots} - R^2_{y\cdot\, jl\ldots}$$

g) ***t-ratio.*** Pode ser usado para testar a hipótese que β, ou B, é igual a zero; isto é, que o preditor i não possui nenhuma influência linear na variável dependente. Sua significância pode ser determinada da tabela de t, com $N - p - 1$ graus de liberdade.

$$t = \left| \frac{\beta_i}{\text{sigma}\,\beta_i} \right| = \left| \frac{B_i}{\text{sigma}\,B_i} \right|$$

h) ***Quociente de covariância.*** O quociente de covariância de x_i é o quadrado do coeficiente de correlação múltipla, R^2, de x_i com as $p-1$ outras variáveis independentes na equação. É uma medida da intercorrelação de x_i com os outros preditores.

$$\text{Quociente de covariancia}_{\,i} = 1 - \frac{1}{c_{ii}}$$

onde c_{ii} é o $i^{'esimo}$ elemento da diagonal da inversa da matriz de correlação dos preditores na equação de regressão (ver a seção 6 acima).

47.9 Resíduos

Os resíduos são a diferença entre o valor observado da variável dependente e o valor predito pela equação de regressão.

$$e_k = y_k - \widehat{y}_k$$

O teste para detectar correlação serial, popularmente conhecido como estatística d de Durbin-Watson para autocorrelação de primeira ordem dos resíduos, é calculada da seguinte forma:

$$d = \frac{\sum_{k=2}^{N} (e_k - e_{k-1})^2}{\sum_{k=1}^{N} e_k^2}$$

47.10 Nota sobre Regressão Stepwise

Regressão stepwise introduz os preditores passo-a-passo no modelo, começando com a variável independente mais correlacionada com y. Depois do primeiro passo, o algoritmo seleciona do grupo de variáveis remanescentes aquela que dá a maior redução na variâncial residual (não-explicada) da variável dependente, i.e. a variável cuja correlação parcial com y é a maior. O programa então executa um teste-F parcial na entrada para checar se a variável absorverá uma quantidade significante da variação em relação aquela removida por variáveis já na regressão. O usuário pode especificar um valor de F mínimo para a inclusão de qualquer variável; o programa avalia se o valor de F obtido em um dado passo satisfaz o mínimo, e se satisfizer, a variável entrará. Similarmente, o programa decide a cada passo se qualquer variável previamente incluída ainda satisfaz um mínimo (também fornecido pelo usuário) e, caso contrário, remove-se tal variável.

$$\text{Valor de F parcial para variável}\, i = \frac{(R^2_{y\cdot\, Pi} - R^2_{y\cdot\, P})(\text{df})}{1 - R^2_{y\cdot\, Pi}}$$

onde

$$\begin{aligned} R^2_{y\cdot Pi} &= R \text{ quadrado múltiplo para o conjunto de preditores } (P) \text{ já na regressão,} \\ &\quad \text{com preditor } i \\ R^2_{y\cdot P} &= R \text{ quadrado múltiplo para o conjunto de preditores } (P) \text{ já na regressão} \\ \text{df} &= \text{graus de liberdade dos resíduos.} \end{aligned}$$

Em qualquer passo do procedimento, os resultados são os mesmos de como seriam em uma regressão padrão usando-se um conjunto particular de variáveis; portanto, o passo final de uma regressão stepwise mostra os mesmos coeficientes que uma execução normal usando as variáveis que "sobreviveram" ao procedimento stepwise.

47.11 Nota sobre Regressão Descendente

Regressão descendente é como regressão stepwise, exceto que o algoritmo começa com todas as variáveis independentes e então elimina e adiciona de volta variáveis de um jeito stepwise.

47.12 Nota sobre Regressão com Intercepto Zero

É possível, ao se usar o programa REGRESSN, requerer-se uma regressão com intercepto zero, i.e. que a variável dependente seja zero quando todas as variáveis independentes forem zero.

Se uma regressão através da origem é especificada, todas as estatísticas, exceto aquelas especificadas nas seções 1 a 4 acima, são baseadas em uma média de zero. O coeficiente de correlação múltipla e a fração de variância explicada (ítens 7.c e 7.d) não são impressos. Estatísticas que não sejam centradas na média podem ser bastante diferentes daquelas que seriam se fosse centradas; portanto, em uma solução stepwise, variáveis podem muito bem entrar na equação em uma ordem diferente daquela que seria se uma constante fosse estimada.

No programa REGRESSN uma matriz com elementos

$$a_{ij} = \frac{\sum_k w_k \, x_{ik} \, x_{jk}}{\sqrt{\sum_k w_k \, x_{ik}^2 \, \sum_k w_k \, x_{jk}^2}}$$

é analisada qo invés de $\mathcal{R}$, a matriz de correlação.

Os B's, coeficientes da regressão parcial não-padronizada, são obtidos por

$$B_i = \beta_i \sqrt{\sum_k w_k \, x_{ik}^2 \, \sum_k w_k \, x_{jk}^2}$$

Capítulo 48

Escalonamento Multidimensional

Notação

x	=	elemento da configuração
i, j, l, m	=	subscritos para variáveis
n	=	número de variáveis
s	=	subscritos para dimensão
t	=	número de dimensões.

48.1 Ordem de Computações

Para um dado número de dimensões, t, MDSCAL encontra a configuração de mínimo esforço pelo uso de um procedimento iterativo. O programa começa com uma configuração inicial (fornecida pelo usuário ou pelo programa) e continua modificando-a até que ela convirja para a configuração de menor esforço.

48.2 Configuração Inicial

Se o usuário não fornecer uma configuração inicial, o programa gera uma configuração arbitrária tomando os primeiros n pontos da seguinte lista (cada expressão entre parênteses representa um ponto):

$$
\begin{array}{c}
(1,0,0,\ldots,0),\\
(0,2,0,\ldots,0),\\
(0,0,3,\ldots,0),\\
\vdots\\
(0,0,0,\ldots,t),\\
(t+1,0,0,\ldots,0),\\
(0,t+2,0,\ldots,0),\\
\vdots
\end{array}
$$

48.3 Centragem e Normalização da Configuração

No início de cada iteração a configuração é centrada e normalizada.

Se x_{is} denota o elemento na $i^{'esima}$ linha e $s^{'esima}$ coluna da configuração, então

$$\text{Centrado } x_{is} = x_{is} - \overline{x}_s$$

$$\text{Normalizado } x_{is} = \frac{x_{is} - \overline{x}_s}{\text{n.f.}}$$

onde

$$\overline{x}_s = \frac{\sum_i x_{is}}{n}$$

é a média da dimensão s e

$$\text{n.f.} \ = \sqrt{\frac{n}{\sum_i \sum_s x_{is}^2}}$$

é o fator de normalização.

Note que a soma total de quadrados dos elementos da configuração centrada normalizada é igual a n, o número de variáveis.

48.4 História de Cálculos

Na conclusão de cada iteração, itens 4.a até 4.h abaixo são impressos. Isso cria uma histórico que, em geral, é de interesse somente quando se tem medo que a convergência não tenha sido completa. Contudo, no final do histórico a razão da parada é impressa. Se o programa não parar porque um mínimo não foi alcançado, pode ser verdade que a solução alcançada seja praticamente indistingüível do mínimo. Mínimo esse que seria alcançado com algumas iterações a mais - em particular, se o esforço é muito pequeno, isso geralmente ocorre.

a) ***Esforço.*** A medida de esforço possui duas funções. Primeiro, ela é uma medida de quão bem a configuração derivada corresponde aos dados de entrada. Segundo, ela é utilizada para decidir que pontos devem continuar na próxima iteração. Há também duas fórmulas disponíveis para o cálculo do esforço: SQDIST e SQDEV.

$$\text{Esforço SQDIST} = \sqrt{\frac{\sum_i \sum_j (d_{ij} - \widehat{d}_{ij})^2}{\sum_i \sum_j d_{ij}^2}}$$

$$\text{Esforço SQDEV} = \sqrt{\frac{\sum_i \sum_j (d_{ij} - \widehat{d}_{ij})^2}{\sum_i \sum_j (d_{ij} - \overline{d}\,)^2}}$$

onde

d_{ij} = distância entre variáveis i e j na configuração (ver 8.c abaixo)

$\widehat{d}_{ij}$ = aqueles números que minimizam o esforço, sujeito a restrição que o d_{ij} tenha a mesma ordem de rank dos dados de entrada (ver 8.d abaixo)

$\overline{d}$ = a média de todos os d_{ij}'s.

b) ***SRAT.*** Quociente de esforço. O usuário pode parar o procedimento de escalonamento especificando o quociente de esforço a ser alcançado. Para a primeira iteração (número 0) seu valor é igualado a 0.800 .

$$\text{SRAT} \ = \frac{\text{Esforço}_{\text{presente}}}{\text{Esforço}_{\text{previo}}}$$

c) ***SRATAV.*** Media do quociente de esforço. Para a primeira iteração seu valor é igual a 0.8000 .

$$\text{SRATAV}_{\text{presente}} = (\text{SRAT}_{\text{presente}})^{0.33334} \ \times \ (\text{SRATAV}_{\text{previo}})^{0.66666}$$

d) ***CAGRGL.*** Esse é o cosseno do ângulo entre o gradiente corrente e o gradiente prévio.

$$\text{CAGRGL} = \cos\Theta = \frac{\sum_i \sum_s g_{is}\, g''_{is}}{\sqrt{\sum_i \sum_s g_{is}^2} \sqrt{\sum_i \sum_s (g''_{is})^2}}$$

onde

$$\begin{aligned} g &= \text{gradiente presente} \\ g'' &= \text{gradiente prévio.} \end{aligned}$$

O gradiente inicial é igualado a uma constante:

$$\text{Inicial } g_{is} = \sqrt{\frac{1}{t}}$$

e) ***COSAV.*** Cosseno médio do ângulo entre gradientes sucessivos. Trata-se de uma média ponderada. Para a primeira iteração, seu valor é igual a 0.

$$\text{COSAV}_{\text{presente}} = \text{CAGRGL}_{\text{presente}} \times \text{COSAVW} + \text{COSAV}_{\text{previo}} \times (1.0 - \text{COSAVW})$$

onde COSAVW é um fator de ponderação sob controle do usuário.

f) ***ACSAV.*** Valor absoluto médio do cosseno do ângulo entre gradientes sucessivos. Trata-se de média ponderada. Para a primeira iteração, seu valor é igual a 0.

$$\text{ACSAV}_{\text{presente}} = |\text{CAGRGL}_{\text{presente}}| \times \text{ACSAVW} + \text{ACSAV}_{\text{previo}} \times (1.0 - \text{ACSAVW})$$

onde ACSAVW é um fator de ponderação sob controle do usuário.

g) ***SFGR.*** Fator de escala do gradiente. À medida que a computação procede, o fator de escala de gradientes sucessivos diminui. Uma maneira do procedimento de escalonamento poder ser parado é alcançar-se um valor mínimo de fator de escala do gradiente.

$$\text{SFGR} = \sqrt{\frac{1}{n} \sum_i \sum_s g_{is}^2}$$

onde g é o gradiente presente.

h) ***STEP.*** Tamanho do passo. Na fórmula de tamanho de passo, os dois principais determinantes do novo tamanho de passo são o passo prévio e o fator de ângulo. Os tamanhos de passo utilizados não afetam a solução final, mas eles afetam o número de iterações requisitadas para alcançar-se uma solução.

$$\text{STEP}_{\text{presente}} = \text{STEP}_{\text{previo}} \times \text{fator de ângulo} \times \text{fator de relaxamento} \times \text{fator de boa sorte}$$

onde

$$\begin{aligned} \text{fator de ângulo} &= 4.0^{\text{COSAV}} \\ \text{fator de relaxamento (ou vies)} &= \frac{1.4}{A\,B} \\ A &= 1 + (\min(1, \text{SRATAV}))^5 \\ B &= 1 + \text{ACSAV} - |\text{COSAV}| \\ \text{fator de boa sorte} &= \sqrt{\min(1, \text{SRAT})} \end{aligned}$$

O primeiro tamanho de passo é computado da seguinte maneira:

$$\text{STEP} = 50. \times \text{Esforço} \times \text{SFGR}$$

48.5 Esforço para Configuração Final

Isso é uma reiteração do último valor da coluna Esforço do histórico de computação (ver 4.a acima). Aqui o Esforço é uma medida de quão bem a configuração final se ajusta aos dados de entrada.

Interpretação do esforço para a configuração final depende da fórmula utilizada nos cálculos. Note que o uso de SQDEV do Esforço produz valores substancialmente maiores de esforço para o mesmo grau de "qualidade de ajuste".

Para o modo clássico de se utilizar MDSCAL, Kruskal e Carmone (1964) oferecem a seguinte tabela de valores de limites de N (digamos, de 10 a 30) e o limite usual de dimensionalidade (digamos, de 2 a 5):

	Esforço SQDIST	Esforço SQDEV
Pobre	20.0 %	40.0 %
Justo	10.0 %	20.0 %
Bom	5.0 %	10.0 %
Excelente	2.5 %	5.0 %
"Perfeito"	0.0 %	0.0 %

48.6 Configuração Final

Em cada iteração a configuração seguinte é formada iniciando-se a partir da configuração antiga e movendo-se ao longo do gradiente (negativo) de esforço uma distância igual ao tamanho de passo.

$$\text{Nova configuração} = \text{configuração antiga} + \frac{\text{PASSO}}{\text{SFGR}}\,(\text{gradiente})$$

Cada linha da matriz de configuração final fornece as coordenadas de uma variável da configuração. A orientação dos eixos de referência é arbitrária e, portanto, deve-se procurar por eixos rotacionados ou mesmo oblíquos que possam ser prontamente interpretáveis. Se uma distância euclidiana ordinária foi usada, é possível rotacionar a configuração de maneira que seus eixos principais coincidam com os eixos de coordenadas. O programa CONFIG pode ser usado para esse propósito.

48.7 Configuração Classificada

Essa é a configuração final apresentada com cada dimensão classificada - as coordenadas são reordenadas da menor para a maior.

48.8 Resumo

a) ***IPOINT, JPOINT.*** São subscritos de variáveis, (i, j), indicando a quais pares de variáveis se referem as três estatísticas abaixo.

b) ***DATA.*** Para cada par de variável, é o índice de entrada de similaridade ou dissimilaridade como fornecido pelo usuário na matriz de dados de entrada.

c) ***DIST.*** Essa é a distância entre pontos na configuração final.

Para a métrica-r de Minkowski,

$$d_{ij} = \left[\sum_s |x_{is} - x_{js}|^r\right]^{1/r}$$

No caso de $r = 2$ isso torna-se a distância euclidiana ordinária

$$d_{ij} = \sqrt{\sum_s (x_{is} - x_{js})^2}$$

No caso de $r = 1$ isso se torna a distância city-block

$$d_{ij} = \sum_s |x_{is} - x_{js}|$$

d) ***DHAT.*** D-hats são números que minimizam o esforço, sujeito a restrição de que d-hats possuem a mesma ordem de rank dos dados de entrada; eles possuem distância "apropriadas", estimadas com os dados de entrada.

Eles são obtidos de

$$\sum_i \sum_j \widehat{d}_{ij} = \sum_i \sum_j d_{ij} \quad \text{e} \quad \widehat{d}_{ij} \geq \widehat{d}_{lm} \quad \text{se} \quad \begin{array}{ll} p_{ij} \leq p_{lm} & \text{(similaridades)} \\ \text{ou} & \\ p_{ij} \geq p_{lm} & \text{(dissimilaridades)} \end{array}$$

onde

d_{ij} = distância entre variáveis i e j na configuração
$\widehat{d}_{ij}$ = uma transformação monotônica dos p_{ij}'s
p_{ij} = o índice de entrada de similaridade ou dissimilaridade entre variáveis i e j.

48.9 Nota sobre Empates nos Dados de Entrada

Empates nos dados de entrada, i.e. valores idênticos na matriz de dados de entrada, podem ser tratados de duas maneiras - a escolha depende do usuário.

A abordagem primária, DIFFER, trata os empates na matriz de entrada com uma relação de ordem indeterminada, que pode ser resolvida arbitrariamente para diminuir dimensionalidade ou esforço.

A abordagem secundária, EQUAL, trata empates como se implicassem uma relação de equivalência, que (tanto quanto possível) deve ser mantida (mesmo se o esforço aumente).

Se houver poucos empates, não faz muita diferença entre qual abordagem escolher.

48.10 Nota sobre Pesos

O programa permite ponderação, mas não é a ponderação no sentido usual do IDAMS.

A ponderação do MDSCAL pode ser usada para designar importância diferente para valores de dados diferentes, isso é, para designar pesos para células da matriz de dados de entrada. Esse tipo de ponderação pode ser usado, por exemplo, para acomodar diferentes variabilidades de medida dentre os valores dos dados.

Se pesos forem utilizados,

$$\text{Esforço SQDIST} = \sqrt{\frac{\sum_i \sum_j w_{ij} \, (d_{ij} - \widehat{d}_{ij})^2}{\sum_i \sum_j w_{ij} \, d_{ij}^2}}$$

$$\text{Esforço SQDEV} = \sqrt{\frac{\sum_i \sum_j w_{ij} \, (d_{ij} - \widehat{d}_{ij})^2}{\sum_i \sum_j w_{ij} \, (d_{ij} - \overline{d} \,)^2}}$$

onde

$$\overline{d} = \frac{\sum_i \sum_j w_{ij} \, d_{ij}}{\sum_i \sum_j w_{ij}}$$

e w_{ij} indica o valor na célula ij da matriz de peso.

48.11 Referências

Kruskal, J.B., Multidimensional scaling by optimizing goodness of fit to a non-metric hypothesis, *Psychometrica*, 3, 1964.

Kruskal, J.B., Nonmetric multidimensional scaling: a numerical method, *Psychometrica*, 29, 1964.

Capítulo 49

Análise de Classificação Múltipla

Notação

y	=	valor da variável dependente
w	=	valor do peso
k	=	subscrito para caso
i	=	subscrito para preditor
j	=	subscrito para categoria de um preditor
p	=	número de preditores
c	=	número de categorias não-vazias ao longo de todos os preditores
a_{ij}	=	desvio ajustado da $j^{'esima}$ categoria do preditor i (ver 2.c abaixo)
N_{ij}	=	número de casos na $j^{'esima}$ categoria do preditor i
N	=	número total de casos
W	=	soma total dos pesos

subscrito ijk indica que o caso k pertence a $j^{'esima}$ categoria do preditor i.

49.1 Estatísticas da Variável Dependente

a) ***Média.*** Média geral de y.

$$\overline{y} = \frac{\sum_k w_k\, y_k}{W}$$

b) ***Desvio-padrão de y (estimado).***

$$\widehat{s}_y = \sqrt{\left(\frac{N}{N-1}\right)\left[\frac{W\sum_k w_k\, y_k^2 - \left(\sum_k w_k\, y_k\right)^2}{W^2}\right]}$$

c) ***Coeficiente de variação.***

$$C_y = \frac{100\,\widehat{s}_y}{\overline{y}}$$

d) ***Soma de y.***

$$\text{Soma de } y = \sum_k w_k\, y_k$$

e) ***Soma de y quadrado.***

$$\text{Soma de } y^2 = \sum_k w_k \, y_k^2$$

f) ***Soma total de quadrados.***

$$\text{TSS} = \sum_k w_k \, (y_k - \overline{y})^2$$

g) ***Soma explicada de quadrados.***

$$\text{ESS} = \sum_i \sum_j a_{ij} \left(\sum_k w_{ijk} \, y_{ijk} \right)$$

h) ***Soma residual de quadrados.***

RSS = TSS - ESS

49.2 Estatísticas de Preditores para Análise de Classificação Múltipla

a) ***Média de classe.*** Média da variável dependente para casos na $j^{'esima}$ categoria do preditor i.

$$\overline{y}_{ij} = \frac{\sum_k w_{ijk} \, y_{ijk}}{\sum_k w_{ijk}}$$

b) ***Desvio não-ajustado em relação a média geral.***

$$\text{Não-ajustado } a_{ij} = \overline{y}_{ij} - \overline{y}$$

c) ***Coeficiente.*** Desvio ajustado a_{ij} em relação a média geral. Esse é o coeficiente de regressão para cada categoria de cada preditor.

$$\text{Predito } y_k = \overline{y} + \sum_i a_{ijk}$$

Os valores de a_{ij} são obtidos por meio de um procedimento iterativo que pára quando $\sum_k (y_k - \textit{predito } y_k)^2$ alcança o mínimo.

d) ***Média de classe ajustada.*** Isso é uma estimativa do que a média deveria ser se o grupo tivesse sido exatamente como a população em termos da distribuição em torno de todas as classificações dos preditores. Se não houvesse nenhuma correlação entre preditores, a média ajustada seria igual a média da classe.

$$\text{Ajustado } \overline{y}_{ij} = \overline{y} + a_{ij}$$

e) ***Desvio-padrão (estimado)*** da variável dependente para a $j^{'esima}$ categoria do preditor i.

$$\widehat{s}_{ij} = \sqrt{\frac{\sum_k w_{ijk} \, y_{ijk}^2 - \left(\sum_k w_{ijk} \, y_{ijk} \right)^2 / \sum_k w_{ijk}}{\sum_k w_{ijk} - \left(\sum_k w_{ijk} / \, N_{ij} \right)}}$$

f) ***Coeficiente de variação (C.var.).***

$$C_{ij} = \frac{100 \, \widehat{s}_{ij}}{\overline{y}_{ij}}$$

g) ***SS de desvios não-ajustados.*** Essa é a soma dos quadrados dos desvios não-ajustados para o preditor i.

$$U_i = \sum_j \Big(\sum_k w_{ijk} \Big) \left(\overline{y}_{ij} - \overline{y} \right)^2$$

h) ***SS de desvios ajustados.*** Essa é a soma dos quadrados dos desvios ajustados para o preditor i.

$$D_i = \sum_j \Big(\sum_k w_{ijk} \Big) \left(a_{ij}^2 \right)$$

i) ***Eta quadrado para o preditor i.*** Eta ao quadrado pode ser interpretado como o percentual de variância na variável dependente que pode ser explicada pelo preditor i somente.

$$\eta_i^2 = \frac{U_i}{\text{TSS}}$$

j) ***Eta para o preditor i.*** Ele indica a habilidade do preditor, usando as categorias dadas para explicar variação na variável dependente.

$$\eta_i = \sqrt{\eta_i^2}$$

k) ***Eta quadrado para preditor i, ajustado pelos graus de liberdade.***

$$\text{Ajustado } \eta_i^2 = 1 - A \left(1 - \eta_i^2 \right)$$

onde A é o ajustamento pelos graus de liberdade (ver 3.b abaixo).

l) ***Eta para o preditor i, ajustado.***

$$\text{Ajustado } \eta_i = \sqrt{1 - A \left(1 - \eta_i^2 \right)}$$

m) ***Beta quadrado para o preditor i.*** Beta ao quadrado é a soma dos quadrados atribuível ao preditor, depois de "manter todos os outros preditores constantes", relativa a soma total de quadrados. Isso *não* significa o percentual da variância explicada.

$$\beta_i^2 = \frac{D_i}{\text{TSS}}$$

n) ***Beta para o preditor i.*** Beta fornece uma medida da habilidade do preditor explicar a variação na variável dependente depois de se ajustar pelos efeitos de todos os outros preditores. Coeficientes beta indicam a importância relativa de vários preditores (quanto maior o valor, mais a variação é explicada pelo beta correspondente).

$$\beta_i = \sqrt{\beta_i^2}$$

49.3 Estatísticas de Análise para Análise de Classificação Múltipla

a) ***R quadrado múltiplo não-ajustado.*** Esse é um coeficiente de correlação múltiplo ao quadrado. Indica a proporção real da variância explicada para os preditores usados na análise.

$$R^2 = \frac{\text{ESS}}{\text{TSS}}$$

b) ***Ajustamento pelos graus de liberdade.***

$$A = \frac{N - 1}{N - p - c - 1}$$

c) ***R quadrado múltiplo ajustado.*** Ele fornece uma estimativa da correlação múltipla na população de onde a amostra é retirada. Note que é uma estimativa da correlação múltipla que seria obtida se os mesmos preditores, mas não necessariamente os mesmos coeficientes, fossem usados para a população.

$$\text{Ajustado } R^2 = 1 - A\,(1 - R^2)$$

d) ***R múltiplo ajustado.*** Esse é o coeficiente de correlação múltipla ajustado pelos graus de liberdade. É uma estimativa do R que seria obtido se os mesmos preditores fosse aplicados à população.

$$\text{Ajustado } R = \sqrt{1 - A\,(1 - R^2)}$$

49.4 Estatísticas de Resumo dos Resíduos

O resíduo para o caso k é $r_k = y_k - predito\ y_k$,

a) ***Média.***

$$\overline{r} = \frac{\sum_k w_k r_k}{W}$$

b) ***Variância (estimada).***

$$\widehat{s}_r^2 = \left(\frac{N}{N-1}\right)\left[\frac{W\sum_k w_k r_k^2 - \left(\sum_k w_k r_k\right)^2}{W^2}\right]$$

c) ***Assimetria.*** A assimetria da distribuição dos resíduos é medida por

$$g_1 = \left(\frac{N}{N-2}\right)\left(\frac{m_3}{\widehat{s}_r^2\sqrt{\widehat{s}_r^2}}\right)$$

onde

$$m_3 = \frac{\sum_k w_k\,(r_k - \overline{r})^3}{W}$$

d) ***Curtose.*** A curtose da distribuição dos resíduos é medida por

$$g_2 = \left(\frac{N}{N-3}\right)\left(\frac{m_4}{(\widehat{s}_r^2)^2}\right) - 3$$

onde

$$m_4 = \frac{\sum_k w_k\,(r_k - \overline{r})^4}{W}$$

49.5 Estatísticas de Categoria do Preditor para Análise de Variância Univariada

Ver o capítulo "Análise de Variância Univariada" para detalhes.

49.6 Estatísticas para Análise de Variância Univariada

Ver o capítulo "Análise de Variância Univariada" para detalhes. Note que o fator de ajustamento A usado no programa MCA para análise de variância univariada é calculado diferentemente do que no programa ONEWAY, de fato:

$$A = \frac{N-1}{N-c}$$

49.7 Referências

Andrews, F.M., Morgan, J.N., Sonquist, J.A., and Klem, L., *Multiple Classification Analysis*, 2nd ed., Institute for Social Research, The University of Michigan, Ann Arbor, 1973.

Capítulo 50

Análise de Variância Multivariada

Notação

y	=	valor da variável dependente ou covariate
i, j	=	subscritos para categorias de preditores
k	=	subscrito para o caso
p	=	número de variáveis dependentes
df_h	=	graus de liberdade para a hipótese
df_e	=	graus de liberdade para o erro.

50.1 Estatísticas Gerais

a) ***Médias das células.*** Faça y_{ijk} representar o valor de uma variável dependente ou covariate para o caso k na subclasse i, j de uma classificação a dois fatores.

$$\overline{y}_{ij} = \frac{\sum_{k=1}^{N_{ij}} y_{ijk}}{N_{ij}}$$

onde N_{ij} é igual ao número de casos na subclasse i, j.

b) ***Base de design.*** A matriz de design é gerada desenvolvendo-se primeiro para cada fator uma matriz de design de um fator (uma matriz K_f) de acordo com o tipo de contraste especificado pelo usuário para aquele fator. A matriz de design geral K é obtida das matrizes de um fator K_f através do produto de Kronecker entre as matrizes.

A matriz de design é sempre impressa com as equações de efeitos em colunas, começando com o efeito da grande média na primeira coluna.

c) ***Intercorrelaçlões entre os coeficientes das equações normais.*** A base do design é ponderada pelas contagens das células. O efeito de frequências das células desiguais é introduzir correlações entre colunas da matriz de design. Essas são aquelas correlações. Se as frequências das células são iguais, aparecerão 1's na diagonal e zeros nas outras posições.

d) ***Solução das equações normais.*** Os parâmetros são estimados via mínimos quadrados na forma

$$LX = (K'DK)^{-1} K'DY$$

onde

L	=	a matriz de contraste que possui como linhas i os contrastes independentes nos parâmetros que deverão ser estimados e testados

X = os parâmetros a serem estimados
K = a matriz de design
D = uma matriz diagonal com o números de casos em cada célula
Y = uma matriz de médias das células com colunas correspondendo a variáveis.

Ao lidar com um design ortogonal e contrastes ortogonais, os contrastes possuem estimativas independentes. Para frequências de células desiguais, no entanto, o K apropriado para designs ortogonais não será mais ortogonal. É necessário transformar K em ortogonalidade na métrica D. Isso é feito colocando

$$T = SK'D^{1/2} \quad \text{com} \quad TT' = T'T = I = SK'DKS'$$

portanto

$$K'D^{1/2} = S^{-1}T$$

e

$$(K'DK)^{-1} = S'S$$

e, substituindo na primeira equação acima,

$$(S')^{-1}LX = SK'DY$$

Essa última equação define um novo conjunto de parâmetros que são funções lineares dos contrastes, com a matriz SK' substituindo K'. Esses parâmetros são ortogonais.

S é uma matriz que calcula a ortogonalização Gram-Schmidt de K na métrica D e reduz as suas linhas ao um comprimento unitário. S, e portanto $(S')^{-1}$, são triangulares.

e) ***Particionamento de matrizes.*** Em uma análise de variância univariada, cada caso possui uma variável dependente y; em uma análise de variância multivariada, cada caso possui um vetor y de variáveis dependentes. O análogo multivariado de y^2 é o produto de matrizes $y'y$ e o análogo multivariado da soma dos quadrados é a soma dos produtos das matrizes.

Em uma análise multivariada, há uma matriz correspondendo a cada soma de quadrados em um design univariado. Testes multivariados dependem das partições da soma total dos produtos, tanto quanto testes univariados dependem das partições da soma total dos quadrados. As fórmulas para a soma total dos produtos, a soma dos produtos entre classes, e a soma de produtos intra classes são

$$S_t = Y'Y$$

$$S_b = Y.'DY.$$

$$S_w = Y'Y - Y.'DY.$$

onde

Y = a matriz original de dados $N \times p$ (N casos, p variáveis dependentes)
$Y.$ = a matriz original de médias das células $n \times p$ (n células, p variáveis dependentes)
D = uma matriz diagonal com o número de casos em cada célula.

A soma dos produtos entre-subclasses é particionado novamente de acordo com os efeitos no modelo.

f) ***Matriz de correlação de erro.*** Em uma análise de variância multivariada, o termo do erro é uma matriz de variância-covariância. Isso é aquele termo de erro reduzido a uma matriz de correlação.

A matriz de correlação é calculado usando S_w, o erro intra, ou a soma dos produtos intra.

$$R_e = s_e^{-1}\, S_w\, s_e^{-1}$$

onde

S_w = a soma dos produtos intra-classes
s_e^2 = os valores da diagonal de S_w.

R_e é a matriz dos coeficientes de correlação das variáveis que estimam os valores populacionais.

Se o usuário especificar que a soma dos quadrados intra-subclasses deve ser aumentado para formar o termo de erro, tal aumento acontece antes da matriz ser reduzida a correlações.

g) ***Componentes principais da matriz de correlação de erro.*** Essa é uma análise padrão de componentes principais da matriz R_e. Indica a estrutura de fatores das variáveis encontradas na população sob estudo. Os valores próprios (ou raízes) são impressos abaixo dos componentes.

h) ***Matriz de dispersão de erro.*** Esse é o termo do erro, uma matriz de variância-covariância, para a análise. A matriz é ajustada para as covariáveis, se houver alguma. Cada elemento da diagonal da matriz será exatamente o que apareceria em uma tabela de análise de variância convencional, como o erro quadrado médio (intra) para a variável.

$$M_e = \frac{S_w}{df_e}$$

onde

S_w = a soma dos produtos intra-subclasses
df_e = os graus de liberdade para o erro, ajustados para o aumento, se isso tiver sido requisitado.

Se o aumento não for requisitado, os graus de liberdade para o erro será igual ao número de casos menos o número de células no design.

i) ***Erros-padrões da estimação.*** Eles correspondem à raiz quadrada dos elementos da diagonal da matriz M_e.

50.2 Cálculos para um Teste em uma Análise Multivariada

Os cálculos são repetidos para cada teste requisitado pelo usuário. Resultados dos cálculos internos descritos abaixo nos pontos a) ao d) não são impressos.

a) ***Matrize de somas de quadrados devido a hipótese.*** A soma dos quadrados entre-subclasses é particionado de acordo com os vários efeitos do modelo. Para uma dada hipótese a ser testada, o programa determina as estimativas ortogonais a serem testadas e computa a soma dos quadrados devido à hipótese (S_h).

b) S_w ***e*** S_h ***reduzidos a quadrados médios e escalonados no espaço de correlação.*** A matriz de quadrados médios para a hipótese, M_h, é calculada analogamente aos quadrados médios do erro.

$$M_h = \frac{S_h}{df_h}$$

onde

S_h = a matriz de somas de quadrados devido a hipótese (ver acima).

Os graus de liberdade para a hipótese dependem do teste requisitado; para um teste de efeito principal A, onde o fator A possui "a" níveis, os graus de liberdade para a hipótese será $a - 1$.

M_h é uma matriz de produtos das médias entre-subclasses associados com o efeito principal da hipótese de interação.

Ambos M_e e M_h são escalonados ao espaço de correlação:

$$R_e = \Delta_e^{-1} M_e \Delta_e^{-1}$$

$$C_h = \Delta_e^{-1} M_h \Delta_e^{-1}$$

onde

R_e = a matriz dos coeficientes de correlação das variáveis que estimam os valores populacionais
C_h = uma matriz que, apesar de não ser uma matriz de correlação, representa as variâncias e covariâncias para as variáveis como afetadas pelo tratamento
M_e = os quadrados médios para o erro
M_h = os quadrados médios para a hipótese
Δ_e = uma matriz diagonal contendo os erros-padrões da estimação.

A matriz R_e é computada duas vezes, uma vez como descrito na seção "Matriz de correlação de erro" e a outra vez como descrito aqui. Se nenhum covariate foi especificado, os resultados são idênticos e a segunda matriz R_e não é impressa. Se um ou mais covariates são especificados, a segunda matriz R_e incorpora ajustamentos para o(s) covariate(s).

c) ***Solução da equação determinadora.*** O método usual para computar o critério da razão de verossimilhança de Wilk advém da equação determinadora

$$|M_h - \lambda M_e| = 0$$

A equação acima é pré e pós-multiplicada pela matriz diagonal Δ_e^{-1}

$$|\Delta_e^{-1} M_h \Delta_e^{-1} - \lambda R_e| = 0$$

Seja

$$R_e = FF'$$

onde

F = a matriz dos coeficientes dos componentes principais satisfazendo $F'F = \omega$, a matriz diagonal de valores próprios de R_e.

A segunda equação determinadora é pré-multiplicada por F^{-1} e pós-multiplicada pela sua transposta, dando

$$|(\Delta_e F)^{-1} M_h ((\Delta_e F)^{-1})' - \lambda F^{-1}(FF')(F^{-1})'| = 0$$

ou

$$|(\Delta_e F)^{-1} M_h ((\Delta_e F)^{-1})' - \lambda I| = 0$$

A última equação é então resolvida para os valores λ.

d) ***Critério de razão de verosimilhança.***

$$\Lambda = \prod_{q=1}^{s} \left(1 + \frac{df_h}{df_e} \times \lambda_q\right)^{-1}$$

onde

λ_q = os valores não nulos oriundos da última equação na seção anterior.

e) ***F-ratio para o critério de razão de verosimilhança.*** O programa usa a aproximação-F dos pontos percentuais da distribuição nula de Λ.

$$F = \frac{1 - \Lambda^{1/k}}{\Lambda^{1/k}} \times \frac{k(2df_e + df_h - p - 1) - p(df_h) + 2}{2p(df_h)}$$

onde

$$k = \sqrt{\frac{p^2(df_h)^2 - 4}{p^2 + (df_h)^2 - 5}}$$

Esse é um teste de significância multivariado do efeito para todas as variáveis dependentes simultaneamente.

f) ***Graus de liberdade do F-ratio.***

$$p(df_h)$$

e

$$\frac{k(2df_e + df_h - p - 1) - p(df_h) + 2}{2}$$

Se $p = 1$ ou 2 e $df_h = 1$ ou 2, k é igualado a 1 nos casos onde $p(df_h) = 2$.

g) ***Variâncias canônicas dos componentes principais da hipótese.*** São os lambdas calculados como descrito na seção "Solução da equação determinadora", acima. Eles são ordenados de maneira decrescente. O número de lambdas que não são zeros para uma dada equação é igual a df_h (o número de graus de liberdade associado com M_h) ou p, o número de variáveis dependentes, o menor desses dois.

h) ***Coeficientes dos componentes principais da hipótese.*** Reesolvendo equação

$$|(\Delta_e F)^{-1} M_h ((\Delta_e F)^{-1})' - \lambda I| = 0$$

dá origem a T, para o qual

$$F^{-1}\ \Delta_e^{-1}\ M_h\ \Delta_e^{-1} (F^{-1})' = T\,\lambda\,T'$$

Isso pode ser reescrito como

$$T'\ F^{-1}\ \Delta_e^{-1}\ X_h\ X_h'\ \Delta_e^{-1} (F^{-1})' T = \lambda$$

A equação acima é considerada como

$$T'\ F^{-1}\ \Delta_e^{-1}\ X_h = S_h^*$$

onde

$$S_h^*\ (S_h^*)' = \lambda$$

e escrita em forma de equação de fator usual, $X = FS$, é

$$\Delta_e^{-1}\ X_h = FTS_h^*$$

Os coeficientes dos componentes principais da hipótese, FT, são impressos pelo programa.

i) ***Escores dos componentes de contraste para efeitos estimados.*** As linhas de S_h^* são conjuntos de escores de fatores, atribuíveis a hipóteses que possuem, como variâncias máximas, o λ_i.

j) ***Testes de Barlett cumulativos nas raízes.*** Os testes podem ser usados para determinar a dimensionalidade da configuração. Os lambdas, ou raízes, são ordenadas de forma crescente de magnitude. Nos testes de Barlett, todas as raízes são testadas primeiramente. Então, todas as outras com exceção da primeira, todas as outras com exceção das duas primeiras, e assim por diante. O teste chi-quadrado fornece um teste de significância para a variância contabilizada pelas $n-k$ raízes depois da aceitação das primeiras k raízes.

Inicialmente os lambdas são escalonados

$$\lambda_i \text{ normatizado } = \frac{df_h}{df_e} \times \lambda_i$$

e então **Chi-quadrado** é calculado

$$\chi^2_{k+1} = \left(df_e + df_h - \frac{df_h + p + 1}{2} \right) \left(\sum_{i=k+1}^{s} \ln(\lambda_i \text{ normatizado } + 1) \right)$$

onde

$$\begin{aligned} k &= \text{o número de raízes aceitas } \ (k = 0, 1, ..., s-1) \\ s &= \text{o número de raízes.} \end{aligned}$$

Os graus de liberdade são

$$DF = (p-k)(g-k-1)$$

onde g é igual ao número de níveis da hipótese.

k) ***F-ratios para testes univariados.*** Esses são os elementos da diagonal de $\Delta_e^{-1} M_h \Delta_e^{-1}$. O F-ratio para a variável y é exatamente o F-ratio que seria obtido para o dado efeito se uma análise univariada fosse executada com a variável y sendo a única variável dependente.

50.3 Análise Univariada

Se uma única variável dependente for especificada, os cálculos, não obstante, são executados como descritos acima. Contudo, há uma vantagem na simplificação, e.g., o componente principal da "matriz" de correlação de erro é igualado a um e nenhum cálculo é executado.

O resultado de uma análise de variância univariada é uma tabela de ANOVA convencional com pequenas diferenças. Ela contém uma linha para média principal, mas não possui uma linha para o total. A média principal é geralmente não interpretável. Para obter-se a soma dos quadrados totais, soma-se todas as somas dos quadrados, exceto a soma da média principal.

50.4 Análise de Covariância

As fórmulas e a discussão acima não consideram, em grande parte, os covariates. Se um ou mais covariates são especificados, são as matrizes de somas de produtos, S_e e S_h que serão ajustadas. Se houver q covariates, o programa inicia-se carregando-os com p variáveis dependentes. Há uma matriz $(p \times q) \times (p \times q)$ de somas de produtos do erro, S_e, e uma matriz $(p \times q) \times (p \times q)$ S_h para cada hipótese. A matriz total S_t é computada. S_e e S_h são particionados em seções correspondendo às variáveis dependentes e covariates. As matrizes reduzidas $(p \times p)$ de erro e total são computadas, e as matrizes reduzidas para hipóteses são então obtidas por subtração.

A matriz de correlação de erro e os componentes principais da matriz são computados depois do ajustamento de S_e para os covariates.

Capítulo 51

Análise de Variância Univariada

Notação

y	=	valor da variável dependente
w	=	valor do peso
k	=	subscrito para caso
i	=	subscrito para categoria da variável de controle
N_i	=	número de casos na categoria i
W_i	=	soma dos pesos para categoria i
N	=	número total de casos
W	=	soma total dos pesos
c	=	número de categorias de código da variável de controle com graus de liberdade não-zero.

51.1 Estatísticas Descritivas para Categorias da Variável de Controle

a) ***Média.***

$$\overline{y}_i = \frac{\sum_k w_{ik}\, y_{ik}}{W_i}$$

b) ***Desvio-padrão (estimado).***

$$\widehat{s}_i = \sqrt{\left(\frac{N_i}{N_i - 1}\right)\left[\frac{W_i \sum_k w_{ik}\, y_{ik}^2 - \left(\sum_k w_{ik}\, y_{ik}\right)^2}{W_i^2}\right]}$$

c) ***Coeficiente de variação (C.var.).***

$$C_i = \frac{100\ \widehat{s}_i}{\overline{y}_i}$$

d) ***Soma de y.***

$$\text{Soma } y_i = \sum_k w_{ik}\, y_{ik}$$

e) ***Percentual.***

$$\text{Percentual}_i = \frac{\text{Soma } y_i}{\sum_i \text{Soma } y_i}$$

f) ***Soma de y quadrado.***

$$\text{Soma } y_i^2 = \sum_k w_{ik}\, y_{ik}^2$$

g) ***Total.*** A linha total fornece as estatísticas 1.a até 1.e acima computadas com todos os casos, exceto em categorias de código com zero graus de liberdade.

h) ***Graus de liberdade para a categoria i.***

$$\text{df}_i = W_i\,(N_i - 1)\,/\,N_i$$

Categorias com zero graus de liberdade não são incluídas na computação das estatísticas de resumo.

51.2 Estatísticas de Análise de Variância

a) ***Soma total de quadrados.***

$$\text{TSS} = \sum_i \sum_k w_{ik}\, y_{ik}^2 \; - \; \frac{\left(\sum_i \sum_k w_{ik}\, y_{ik}\right)^2}{W}$$

b) ***Soma de quadrados entre médias.*** Isso é, às vezes, chamado de a "soma de quadrados entre grupos" (ou inter-grupos).

$$\text{BSS} = \sum_i \left[\frac{\left(\sum_k w_{ik}\, y_{ik}\right)^2}{\sum_k w_{ik}}\right] - \frac{\left(\sum_i \sum_k w_{ik}\, y_{ik}\right)^2}{W}$$

c) ***Soma de quadrados dentro dos grupos.*** Isso é, às vezes, chamado de a "soma de quadrados intra-grupos".

$$\text{WSS} = \text{TSS - BSS}$$

d) ***Eta quadrado.*** Essa medida pode ser interpretada como o percentual da variância na variável dependente que pode ser explicado pela variável de controle. Ele assume valores de 0 a 1.

$$\eta^2 = \frac{\text{BSS}}{\text{TSS}}$$

e) ***Eta.*** Essa é uma medida da intensidade da associação entre a variável dependente e a variável de controle. Ela varia de 0 a 1.

$$\eta = \sqrt{\frac{\text{BSS}}{\text{TSS}}}$$

f) ***Eta quadrado ajustado.*** Eta quadrado ajustado pelos graus de liberdade.

$$\text{Ajustado } \eta^2 = 1 - A(1 - \eta^2)$$

com fator de ajustamento

$$A = \frac{W - 1}{W - c}$$

g) ***Eta ajustado.***

$$\text{Ajustado } \eta = \sqrt{\text{Ajustado } \eta^2}$$

h) ***F-ratio.*** O F-ratio diz respeito à distribuição F com $c-1$ e $N-c$ graus de liberdade. Um F significante significa que as diferenças de média, ou efeitos, provavelmente existe entre os grupos.

$$F = \frac{\text{BSS}/(c-1)}{\text{WSS}/(N-c)}$$

O F-ratio não é computado se uma variável de peso foi especificada.

Capítulo 52

Scoring Baseado em Ordenação Parcial de Casos

52.1 Terminologia Especial e Definições

Denote um conjunto de elementos por $\mathcal{V} = \{a, b, c, \dots, \}$ e uma relação binária definida nesse conjunto por R.

a) ***Relação binária.*** Uma relação binária R em $\mathcal{V}$ é definida de tal maneira que, para quaisquer dois elementos $a, b \in \mathcal{V}$,

$$aRb$$

Para qualquer relação binária R em $\mathcal{V}$ existe uma relação *conversa* R^+ em $\mathcal{V}$ tal que

$$bR^+a$$

b) ***Relação reflexiva e anti-reflexiva.*** A relação R é reflexiva quando

$$aRa \qquad \text{para todo } a \in \mathcal{V}$$

e R é anti-reflexiva

$$not(aRa) \qquad \text{para todo } a \in \mathcal{V}$$

c) ***Relação simétrica e anti-simétrica.*** Uma relação é simétrica quando $R = R^+$, isto é quando

$$aRb \Longleftrightarrow bRa \qquad \text{para todo } a, b \in \mathcal{V}$$

e R é anti-simétrica quando a simetria não aparece para todos os $a \neq b$.

d) ***Relação transitiva.*** Uma relação R é transitiva quando

$$aRb \wedge bRc \Longrightarrow aRc \qquad \text{para todo } a, b, c \in \mathcal{V}$$

e) ***Relação de equivalência.*** Uma relação R definida em um conjunto de elementos $\mathcal{V}$ é uma relação de equivalência quando ela é:

- reflexiva,
- simétrica, e
- transitiva.

Note que a comumente usada relação de "igualdade", $(=)$, definida no conjunto dos números reais, é uma relação de equivalência.

f) ***Relação de ordem parcial estrita.*** Uma relação R é chamada de ordem parcial estrita quando satisfizer as seguintes condições:

- aRb e bRa não pode ocorrer simultaneamente, e
- R é transitiva.

Uma relação de ordem parcial estrita é denotada daqui em diante por $\prec$.

g) ***Conjunto parcialmente ordenado.*** Um conjunto $\mathcal{V}$ é chamado de parcialmente ordenado se uma relação de ordem parcial estrita "$\prec$" é definida nele. As propriedades fundamentais de um conjunto parcialmente ordenado são:

- $a \prec b \wedge b \prec c \Longrightarrow a \prec c \qquad$ para todo $a, b, c \in \mathcal{V}$
- $a \prec b$ e $b \prec a$ não pode ocorrer simultaneamente.

h) ***Conjunto ordenado.*** Um conjunto $\mathcal{V}$ é dito um conjunto ordenado se houver duas relações "$\approx$" e "$\prec$" definidas nesse conjunto e se elas satisfizerem os axiomas de ordenamento:

- para quaisquer dois elementos $a, b \in \mathcal{V}$, uma e somente uma das relações $a \approx b$, $a \prec b$, $b \prec a$ aplica-se,
- "$\approx$" é uma relação de equivalência, e
- "$\prec$" é uma relação transitiva.

Em outras palavras, um conjunto ordenado é um conjunto parcialmente ordenado com relações de equivalência adicionais definidas, e onde as condições "nem $a \prec b$ nem $b \prec a$" e "$a \approx b$" são equivalentes.

i) ***Subconjunto de elemetos dominando um elemento a.***

$$\mathcal{G}(a) = \left\{ g \mid g \in \mathcal{V};\ a \prec g \right\}$$

j) ***Subconjunto de elementos dominados por um elemento a.***

$$\mathcal{L}(a) = \left\{ l \mid l \in \mathcal{V};\ l \prec a \right\}$$

k) ***Subconjunto de elementos comparáveis.***

$$\mathcal{C}(a) = \mathcal{G}(a) \cup \mathcal{L}(a)$$

Note que $\mathcal{G}(a) \cap \mathcal{L}(a) = \emptyset$.

l) ***Dominância estrita.*** Um elemento b domina estritamente um elemento a se

$$a \prec b \quad \text{e} \quad \text{not}(b \prec a)$$

Pode-se dizer também que "b é estritamente melhor que a", ou que "a é estritamente pior que b".

52.2 Cálculo dos Escores

Denote uma lista de variáveis a ser usada na análise por

$$\{x_1, x_2, \ldots, x_i, \ldots, x_v\}$$

e uma lista de prioridade associada a elas por

$$\{p_1, p_2, \ldots, p_i, \ldots, p_v\}.$$

A RELAÇÃO DE ORDEM PARCIAL construída com base nessa coleção de variáveis,

$a \prec b \quad$ para quaisquer casos a e b

é equivalente à condição

$$x_1(a) \leq x_1(b),\ x_2(a) \leq x_2(b),\ \ldots,\ x_v(a) \leq x_v(b)$$

onde $x_i(a)$ e $x_i(b)$ denota valores da i^{esima} variável para casos a e b, respectivamente.

Ao COMPARAR-SE DOIS CASOS, as variáveis com as maiores prioridades (menor valor de LEVEL) são consideradas primeiramente. Se elas determinam a relação sem ambigüidade, o procedimento de comparação termina. No caso de igualdade, a comparação é continuada usando-se vaiáveis do próximo nível de prioridade. Esse procedimento repetido até que a relação seja determinada em um dos níveis de prioridade, ou até que o fim da lista de variáveis seja alcançado.

Para cada caso a do conjunto analisado, o programa calcula:

$$\begin{aligned}
\overline{N}(a) &= \text{o número de casos estritamente dominando o caso } a \\
N(a) &= \text{o número de casos equivalentes ao caso } a \\
\underline{N}(a) &= \text{o número de casos estritamente dominados pelo caso } a
\end{aligned}$$

e então um (ou dois) dos seguintes escores:

$$s_1(a) = S\,\frac{\underline{N}(a)}{\underline{N}(a) + N(a) + \overline{N}(a)}$$

$$r_1(a) = S - s_1(a)$$

$$s_2(a) = S\,\frac{\underline{N}(a) + N(a)}{\underline{N}(a) + N(a) + \overline{N}(a)}$$

$$r_2(a) = S - s_2(a)$$

$$s_3(a) = S\,\frac{\underline{N}(a)}{N}$$

$$r_3(a) = S\,\frac{\overline{N}(a) + N(a)}{N}$$

$$s_4(a) = S\,\frac{\underline{N}(a) + N(a)}{N}$$

$$r_4(a) = S\,\frac{\overline{N}(a)}{N}$$

onde

$$\begin{aligned}
N &= \text{número total de casos no conjunto analisado} \\
S &= \text{o valor do fator de escala (ver o parâmetro SCALE).}
\end{aligned}$$

Os valores do parâmetro ORDER seleciona os escores da seguinte maneira:

$$\begin{aligned}
\text{ASEA} &: r_3(a) \\
\text{DEEA} &: s_4(a) \\
\text{ASCA} &: r_4(a) \\
\text{DESA} &: s_3(a) \\
\text{ASER} &: s_1(a), r_1(a) \\
\text{DESR} &: s_1(a), r_1(a) \\
\text{ASCR} &: s_2(a), r_2(a) \\
\text{DEER} &: s_2(a), r_2(a).
\end{aligned}$$

52.3 Referências

Debreu, G., Representation of a preference ordering by a numerical function, *Decision Process*, eds. R.M. Thrall, C.A. Coombs and R.L. Davis, New York, 1954.

Hunya, P., *A Ranking Procedure Based on Partially Ordered Sets*, Internal paper, JATE, Szeged, 1976.

Capítulo 53

Correlação de Pearson

Notação

x, y	=	valores de variáveis
w	=	valor do peso
k	=	subscrito para caso
N	=	número de casos válidos em ambos x e y
W	=	soma total dos pesos.

53.1 Estatísticas Emparelhadas

Elas são computadas para variáveis tomadas em pares (x, y) no subconjunto de casos possuindo dados válidos em ambos x e y.

a) ***Soma ponderada ajustada.*** O número de casos, ponderados, com dados válidos em ambos x e y.

b) ***Média de x.***

$$\overline{x} = \frac{\sum_k w_k \, x_k}{W}$$

Note: a fórmula para *média de y* é análoga.

c) ***Desvio-padrão de x (estimado).***

$$\widehat{s}_x = \sqrt{\left(\frac{N}{N-1}\right)\left[\frac{W \sum_k w_k \, x_k^2 - \left(\sum_k w_k \, x_k\right)^2}{W^2}\right]}$$

Note: a fórmula para *desvio-padrão de y* é análoga.

d) ***Coeficiente de correlação.*** Coeficiente do momento de produto de Pearson r.

$$r_{xy} = \frac{W \sum_k w_k \, x_k \, y_k - \left(\sum_k w_k \, x_k\right)\left(\sum_k w_k \, y_k\right)}{\sqrt{\left[W \sum_k w_k \, x_k^2 - \left(\sum_k w_k \, x_k\right)^2\right]\left[W \sum_k w_k \, y_k^2 - \left(\sum_k w_k \, y_k\right)^2\right]}}$$

e) ***Teste t.*** Essa estatística é usada para testar a hipótese de que o coeficiente de correlação da população é zero.

$$t = \frac{r\sqrt{N-2}}{\sqrt{1-r^2}}$$

53.2 Médias e Desvios-Padrões Não-emparelhadas

Elas são computadas, variável por variável, para todas as variáveis inclusas na análise, usando fórmulas dadas em 1.a, 1.b e 1.c, respectivamente, a diferença potencial nos resultados devendo-se ao número diferente de casos válidos.

a) ***Soma ponderada ajustada.*** O número de casos, ponderados, com dados válidos em x.

b) ***Média de x.*** Média da variável x para todos os casos com dados válidos em x.

c) ***Desvio-padrão de x (estimado).*** Desvio-padrão da variável x para todos os casos com dados válidos em x.

53.3 Equação de Regressão para Escores Brutos

É computada em todos os casos válidos para o par (x, y).

a) ***Coeficiente de regressão.*** Trata-se do coeficiente de regressão não-padronizado de y (variável dependente) em x (variável dependente).

$$B_{yx} = r_{xy}\left(\frac{\widehat{s}_y}{\widehat{s}_x}\right)$$

b) ***Termo constante.***

$$A = \overline{y} - B_{yx}\,\overline{x}; \quad \text{equação de regressão:} \quad y = B_{yx}\,x + A$$

53.4 Matriz de Correlação

Os elementos dessa matriz são computados tomando como base a fórmula dada em 1.d acima. Note que os desvios-padrões produzidos com a matriz de correlação são calculados de acordo com a fórmula dada em 1.c acima (desvios-padrões estimados).

53.5 Matriz de Produtos Cruzados

É uma matriz quadrada com os seguintes elementos:

$$\mathrm{CP}_{xy} = \sum_k w_k\,x_k\,y_k$$

53.6 Matriz de Covariância

É uma matriz contendo os seguinte elementos:

$$\mathrm{COV}_{xy} = r_{xy}\,s_x\,s_y$$

onde

$$s_x = \sqrt{\frac{W \sum_k w_k \, x_k^2 - \left(\sum_k w_k \, x_k \right)^2}{W^2}}$$

e s_y são calculados de acordo com as fórmulas análogas.

Note que a matriz de covariância produzida por PEARSON não contém elementos da diagonal. Para permitir serem recalculados, desvios-padrões produzidos com essa matriz são calculados de acordo com a fórmula acima (desvios-padrões não-estimados).

Capítulo 54

Ordenamento de Alternativas

Notação

i, j, l	=	subscritos para alternativas
m	=	número de alternativas
k	=	índice de caso
n	=	número de casos
w	=	valor do peso.

54.1 Manuseamento dos Dados de Entrada

Seja um CONJUNTO DE ALTERNATIVAS denotado por $\mathcal{A} = \{a_1, a_2, \ldots, a_i, \ldots, a_m\}$ e o conjunto de fontes de informação (chamado, daqui em diante, de AVALIAÇÕES) denotado por $\mathcal{E} = \{e_1, e_2, \ldots, e_k, \ldots, e_n\}$.

Na prática, dados fornecendo a informação primária na relação de preferências podem aparecer em várias formas diferentes. O programa aceita, contudo, dois tipos básicos de dados: dados representando uma seleção de alternativas e dados representando um ordenamento de alternativas. Todas as outras formas devem ser transformadas pelo usuário antes da execução do programa RANK.

a) ***Dados representando uma seleção de alternativas.*** Nesse caso as avaliações representam a escolha das alternativas mais preferidas e opcionalmente suas ordens de preferências. Em outras palavras, todas as avaliações e_k selecionam um subconjunto $\mathcal{A}_k$ de $\mathcal{A}$ e opcionalmente ordenam os elementos dele. Por esse motivo $\mathcal{A}_k$ é um subconjunto de alternativas (ordenado ou não-ordenado), e o $\mathcal{A}_k$'s constituem os dados primários individuais:

$$\mathcal{A}_k = \left\{ a_{ki_1}, a_{ki_2}, \ldots, a_{ki_{p_k}} \right\}$$

onde

p	=	número máximo de alternativas que podem ser selecionadas em uma avaliação
p_k	=	número de alternativas realmente selecionadas na avaliação e_k

e $p_k \leq p < m$.

b) ***Dados representando um ordenamento de alternativas.*** Aqui as avaliações representam o ordenamento de alternativas dentro do conjunto completo $\mathcal{A}$, e a atribuição a cada uma delas do número de rango. Formalmente, todas as avaliações e_k fornecem um número de rank $\rho_k(a_i) = \rho_{ki}$ para todas as alternativas. Nesse caso os dados são fornecidos no seguinte formato:

$$\mathcal{P}_k = \{\rho_k(a_1), \rho_k(a_2), \ldots, \rho_k(a_m)\}$$

Note que a alternativa a_{ki_1} "é estritamente melhor que" ou "domina estritamente" outra alternativa a_{ki_2} de acordo com os dados advindos da avaliação e_k se o primeiro tiver um rango maior do que o do segundo. Similarmente, um alternativa a_{ki_1} "é preferível a" ou "domina" outra alternativa a_{ki_2} de acordo com os dados advindos da avaliação e_k se o rank de a_{ki_1} é pelo menos tão alto quanto o rank de a_{ki_2}. O valor "1" é designado para o maior rango.

Somente os dados descritos no parágrafo b) são diretamente processados pelo programa. Os dados descritos em a) são transformados no formato de b). Essa transformação faz uma distinção entre a preferência estrita e fraca.

A REGRA DE TRANSFORMAÇÃO, ao se lidar com dados representando uma seleção completamente ordenada de alternativas (preferência estrita), é a seguinte:

$$\begin{aligned} &\text{para } a_i \in \mathcal{A}_k \qquad \rho_k(a_{i_1}) = 1,\ \rho_k(a_{i_2}) = 2,\ \ldots,\ \rho_k(a_{ip_k}) = p_k \\ &\text{para } a_i \notin \mathcal{A}_k \qquad \rho_k(a_i) = \frac{p_k + 1 + m}{2} \end{aligned}$$

Ao lidar-se com dados representando uma seleção não-ordenada de alternativas (preferência fraca), é assumido que todas as alternativas selecionadas estão a um mesmo nível de preferência. De acordo com essa hipótese, a regra de transformação é:

$$\begin{aligned} &\text{para } a_i \in \mathcal{A}_k \qquad \rho_k(a_i) = \frac{p_k + 1}{2} \\ &\text{para } a_i \notin \mathcal{A}_k \qquad \rho_k(a_i) = \frac{p_k + 1 + m}{2} \end{aligned}$$

Como resultado das transformações definidas acima, os dados de preferência (ou escolha de prioridade) devem, para os próximos passos da análise, vir na foma:

$$\mathcal{P}_{(n,m)} = \begin{bmatrix} \rho_{11} & \rho_{12} & \cdots & \rho_{1i} & \cdots & \rho_{1m} \\ \rho_{21} & \rho_{22} & \cdots & \rho_{2i} & \cdots & \rho_{2m} \\ \vdots & \vdots & & \vdots & & \vdots \\ \rho_{k1} & \rho_{k2} & \cdots & \rho_{ki} & \cdots & \rho_{km} \\ \vdots & \vdots & & \vdots & & \vdots \\ \rho_{n1} & \rho_{n2} & \cdots & \rho_{ni} & \cdots & \rho_{nm} \end{bmatrix}$$

54.2 Método Baseado em Lógica Clássica

Nesse método a matriz $\mathcal{P}$ é usada como dado inicial para a análise. No que diz respeito à característica estrita ou fraca da relação de preferências, deve ser notado que ela é importante somente nos passos que levam à matriz $\mathcal{P}$. Nos passos seguintes da análise, o procedimento é contolado por outros parâmetros, como a diferença de ranks para concordância e a diferença de ranks para discordância (ver abaixo).

O procedimento de ordenamento baseado em lógica clássica consiste de dois grandes passos: a) construção das relações, e b) identificação dos núcleos.

a) ***Construção das relações.*** Nesse passo, duas relações "que funcionem" (a relação de concordância e a relação de discordância) são contruídas primeiramente. Depois, elas são usadas para construir uma relação de dominância final.

i) AS RELAÇÕES DE CONCORDÂNCIA E DE DISCORDÂNCIA são construídas da matriz $\mathcal{P}_{(n,m)}$, e as regras aplicadas nesse processo são essencialmente as mesmas para ambs as relações.

RELAÇÃO DE CONCORDÂNCIA. Dois parâmetros são utilizados para criar-se uma relação que reflita a concordância da opinião coletiva que "a_i é preferível a a_j":

$$\begin{aligned} d_c &= \text{a diferença de ranks para concordância } \ (0 \le d_c \le m-1) \\ p_c &= \text{a proporção mínima para concordância } \ (0 \le p_c < 1). \end{aligned}$$

A diferença de ranks para concordância permite ao usuário influenciar a avaliação dos dados

quando da construção das matrizes de preferências individuais

$$\mathrm{RC}^k(d_c) = \left[\mathrm{rc}^k_{ij}(d_c) \right] \quad \text{onde } \; i,j = 1,2,\ldots,m.$$

Os elementos de $\mathrm{RC}^k(d_c)$, que medem a dominância de a_i sobre a_j de acordo com a avaliação k, são definidos da seguinte maneira:

$$\mathrm{rc}^k_{ij}(d_c) = \begin{cases} 1 & \text{se } \; \rho_{kj} - \rho_{ki} \geq d_c \\ 0 & \text{caso contrario.} \end{cases}$$

A agregação dessas matrizes mede a dominância média de a_i sobre a_j e possui a forma de uma relação difusa descrita pela matriz

$$\mathrm{RC}(d_c) = \left[\mathrm{rc}_{ij}(d_c) \right]$$

onde

$$\mathrm{rc}_{ij}(d_c) = \frac{\sum\limits_k w_k \, \mathrm{rc}^k_{ij}(d_c)}{\sum\limits_k w_k}$$

Note que maiores valores de d_c levam a regras de construção mais rigorosas, pois $\; d^1_c < d^2_c$ implica

$$\mathrm{rc}^k_{ij}(d^1_c) \; \geq \; \mathrm{rc}^k_{ij}(d^2_c) \quad \text{ e } \quad \mathrm{rc}_{ij}(d^1_c) \; \geq \; \mathrm{rc}_{ij}(d^2_c)$$

Proporção mínima para concordância torna possível transformar a relação difusa $\mathrm{RC}(d_c)$ em uma relação não-difusa, chamada a relação de concordância, descrita pela matriz

$$\mathrm{RC}(d_c, p_c) = \left[\mathrm{rc}_{ij}(d_c, p_c) \right]$$

cujos elementos são definidos da seguinte maneira:

$$\mathrm{rc}_{ij}(d_c, p_c) = \begin{cases} 1 & \text{se } \; \mathrm{rc}_{ij}(d_c) \geq p_c \\ 0 & \text{caso contrario.} \end{cases}$$

A condição $\mathrm{rc}_{ij}(d_c, p_c) = 1$ significa que a opinião coletiva está em concordância com a declaração "a_i é preferível a a_j" ao nível (d_c, p_c).

Fica claro novamente que aumentando o valor de p_c obtem-se condições estritas para concordância.

Relação de discordância. A construção da relação de discordância segue a mesma maneira como explicado no caso da concordância. Os dois parâmetros controlando a construção são:

$$\begin{aligned} d_d &= \text{a diferença de ranks para discordância } \; (0 \leq d_d \leq m-1) \\ p_d &= \text{a proporção máxima para discordância } \; (0 \leq p_d \leq 1). \end{aligned}$$

As relações de discordância individuais são determinadas primeiramente nas matrizes

$$\mathrm{RD}^k(d_d) = \left[\mathrm{rd}^k_{ij}(d_d) \right] \quad \text{ onde } \quad i,j = 1,2,\ldots,m.$$

Os elementos de $\mathrm{RD}^k(d_d)$, que medem a dominância de a_j sobre a_i de acordo com a avaliação k, são definidos da seguinte maneira:

$$\mathrm{rd}^k_{ij}(d_d) = \begin{cases} 1 & \text{se } \; \rho_{ki} - \rho_{kj} \geq d_d \\ 0 & \text{caso contrario.} \end{cases}$$

A agregação dessas matrizes mede a dominância média de a_j sobre a_i e possui a forma de uma relação difusa descrita pela matriz

$$\mathrm{RD}(d_d) = \left[\mathrm{rd}_{ij}(d_d) \right]$$

onde

$$\mathrm{rd}_{ij}(d_d) = \frac{\sum\limits_k w_k \, \mathrm{rd}^k_{ij}(d_d)}{\sum\limits_k w_k}$$

Para o caso da concordância, o segundo parâmetro (proporção máxima para discordância), possibilita ao usuário transformar a relação difusa $\mathrm{RD}(d_d)$ em uma relação não-difusa, chamada de relação de discordância, descrita pela matriz

$$\mathrm{RD}(d_d, p_d) = \Big[\ \mathrm{rd}_{ij}(d_d, p_d)\ \Big]$$

cujos elementos são definidos da seguinte maneira:

$$\mathrm{rd}_{ij}(d_d, p_d) = \begin{cases} 1 & \text{se } \ \mathrm{rd}_{ij}(d_d) > p_d \\ 0 & \text{caso contrario.} \end{cases}$$

A condição $\mathrm{rd}_{ij}(d_d, p_d) = 1$ significa que a opinião coletiva está em discordância com a declaração "a_i é preferível a a_j", i.e. suporta a declaração "a_j é preferível a a_i", ao nível (d_d, p_d). Isso pode ser interpretado como um "veto coletivo" contra a declaração "a_i é preferível a a_j".

Note que maiores valores de d_d e p_d levam a regras de construção menos rigorosas e, portanto, a condições para discordância mais fracas.

ii) A RELAÇÃO DE DOMINÂNCIA é composta das relações de discordância e de concordância. A idéia básica é que a declaração "a_i é preferível a a_j" possa ser aceita se a opinião coletiva

- está em concordância com isso, i.e. $\mathrm{rc}_{ij}(d_c, p_c) = 1$, e
- não está em discordância com isso, i.e. $\mathrm{rd}_{ij}(d_d, p_d) = 0$;

caso contrário, essa declaração deve ser rejeitada. De fato, a relação de dominância, sendo uma função de quatro parâmetros, é descrita pela matriz $\mathcal{R}$ de $m \times m$ dimensões

$$\mathcal{R} = \Big[\ r_{ij}(d_c, p_c, d_d, p_d)\ \Big]$$

onde os elementos são obtidos de acordo com a expressão

$$r_{ij}(d_c, p_c, d_d, p_d) = \min\big[\ \mathrm{rc}_{ij}(d_c, p_c),\ 1 - \mathrm{rd}_{ij}(d_d, p_d)\ \big]$$

O r_{ij} é uma função monotonicamente decrescente dos dois primeiros parâmetros, e uma função monotonicamente crescente dos dois últimos. Isso implica que:

- ao aumentar-se o d_c, p_c e/ou diminuir-se d_d, p_d, pode-se diminuir o número de conecções na relação de dominância, e
- ao mudar-se os parâmetros na direção oposta, pode-se criar mais conecções.

b) ***Identificação de núcleos.*** Os NÚCLEOS são subconjuntos de $\mathcal{A}$ (conjunto de alternativas) consistindo de alternativas não-dominadas. Uma alterativa a_j é não-dominada se, e somente se

$$r_{ij} = 0 \quad \text{para todo } i = 1, 2, \ldots, m.$$

i) De acordo com esse critério, o núcleo do conjunto $\mathcal{A}$ (o núcleo de maior nível) é o subconjunto

$$\mathcal{C}(\mathcal{A}) = \Big\{\ a_j \mid a_j \in \mathcal{A};\ r_{ij} = 0,\ i = 1, 2, \ldots, m\ \Big\}$$

- Se $\mathcal{C}(\mathcal{A}) = \emptyset$ então todas as alternativs são dominadas.
- Se $\mathcal{C}(\mathcal{A}) = \mathcal{A}$ então todas as alternativas são não-dominadas.

ii) Para encontrar o núcleo subseqüente, os elementos do núcleo anterior são removidos da relação de dominância primeiramente. Isso significa que as linhas e colunas correspondentes são removidas da matriz relacional. Então, a busca por um novo núcleo é repetida em na forma reduzida.

A aplicação sucessiva de i) e ii) fornece uma série de núcleos $\mathcal{A}_1^c$, $\mathcal{A}_2^c$, $\ldots$, $\mathcal{A}_q^c$. Esses núcleos representam camadas consecutivas de alternativas com ranges decrescentes na estrutura de preferências, enquanto as alternativas pertencendo a um mesmo núcleo são assumidas possuirem um mesmo rango.

54.3 Métodos Baseados em Lógica Difusa: A Relação de Entrada

Nos métodos de ordenamento baseados em lógica difusa, a matriz $\mathcal{P}_{(n,m)}$ é usada para construir: a) relações de preferências individuais, e b) a relação de entrada (chamada também de "relação difusa") no conjunto de alternativas $\mathcal{A}$. Aqui a característica estrita ou fraca da relação de preferências possui um papel importante.

a) ***Construção das relações de preferências individuais.*** Para cada avaliação e_k uma relação de preferência individual, que é implicitamente dada em $\mathcal{P}$, é transformada em uma matriz de $m \times m$ dimensões:

$$\mathcal{R}^k = \left[r_{ij}^k \right] \quad \text{onde } i, j = 1, 2, \ldots, m$$

onde

$$r_{ij}^k = \begin{cases} 1 & \text{se a frase "}a_i \text{ e preferível a } a_j \text{ na avaliação } e_k\text{" e verdade;} \\ 0 & \text{se a frase e falsa.} \end{cases}$$

Dependendo do tipo de preferência utilizada, a declaração "a_i é preferível a a_j na avaliação e_k" é equivalente a desigualdade

$$\begin{array}{ll} \rho_{ki} < \rho_{kj} & \text{(preferência estrita), ou} \\ \rho_{ki} \leq \rho_{kj} & \text{(preferência fraca).} \end{array}$$

b) ***Construção da relação de entrada (relação difusa).*** A agregação de matrizes de relação de preferências individuais fornece uma matriz representando uma relação difusa no conjunto de alternativas $\mathcal{A}$:

$$\mathcal{R} = \left[r_{ij} \right]$$

onde

$$r_{ij} = \frac{\sum\limits_k w_k \, r_{ij}^k}{\sum\limits_k w_k}$$

Cada componente r_{ij} de $\mathcal{R}$ pode ser interpretado como a credibilidade das declarações "a_i é preferível a a_j" em um sentido global, e sem se referir a nenhuma avaliação. Portanto, a seguinte interpretação geral é possível:

$$\begin{array}{ll} r_{ij} = 1 & \text{"}a_i \text{ e preferível a } a_j\text{" em todas avaliações,} \\ r_{ij} = 0 & \text{"}a_i \text{ e preferível a } a_j\text{" em nenhuma avaliação,} \\ 0 < r_{ij} < 1 & \text{"}a_i \text{ e preferível a } a_j\text{" em uma certa porção de avaliações.} \end{array}$$

c) ***Características da relação de entrada.***

i) Fuzzyness

não-difusa : se $r_{ij} = 0$ ou $r_{ij} = 1$ para todo $i, j = 1, 2, \ldots, m$;
difusa : caso contrario.

ii) Simetria

simétrica : se$r_{ij} = r_{ji}$ para todo $i, j = 1, 2, \ldots, m$;
anti-simétrica : se $r_{ij} \neq 0$ implica $r_{ji} = 0$ para todo $i \neq j$;
assimétrica : demais casos.

iii) Reflexividade

reflexiva : se $r_{ii} = 1$ para todo $i = 1, 2, \ldots, m$;
anti-reflexiva : se $r_{ii} = 0$ para todo $i = 1, 2, \ldots, m$;
irreflexiva : demais casos.

iv) Tricotomia

tricótoma : se $r_{ij} + r_{ji} = 1$ para todo $i, j = 1, 2, \ldots, m$ e $i \neq j$;
(normalizado)
não-tricótoma : demais casos.
(não-normalizado)

v) Índice de coerência. Seu valor, C, depende da ordem das linhas e colunas em $\mathcal{R}$, i.e. na ordem das alternativas em $\mathcal{A}$, e $-1 \leq C \leq 1$.

$$C = \frac{\sum_{i<j}(r_{ij} - r_{ji})}{\sum_{i<j}(r_{ij} + r_{ji})}$$

Índice de coerência absoluta é uma modificação independente de ordem de C. Seu valor, C_a, é o limite superior para C e $0 \leq C_a \leq 1$.

$$C_a = \frac{\sum_{i<j}|r_{ij} - r_{ji}|}{\sum_{i<j}(r_{ij} + r_{ji})}$$

Os índice C e C_a são indicadores da unanimidade nos dados de preferência. Uma coerência completa é mostrada quando $C = 1$, enquanto $C_a = 0$ indica uma completa ausência de coerência. O valor -1 para o índice C pode ser interpretado como uma ordem de alternativas oposta a ordem definida pela relação difusa.

vi) Índice de intensidade. Esse índice pode ser interpretado como um nível de credibilidade média das declarações "a_i é preferível a a_j" ou "a_j é preferível a a_i". Em geral, seu valor $-1 \leq I \leq 2$, enquanto no caso de uma preferência estrita $0 \leq I \leq 1$. Aqui $I = 1$ implica uma relação normalizada (ver seção 3.c abaixo) e significa que em todos os dados de preferência, uma das declarações acima é válida para todos os pares de alternativas.

$$I = \frac{\sum_{i<j}(r_{ij} + r_{ji})}{m(m-1)/2}$$

vii) Índice de dominância. É também um índice dependente da ordem, e $-1 \leq D \leq 1$.

$$D = \frac{\sum_{i<j}(r_{ij} - r_{ji})}{m(m-1)/2}$$

Índice de dominância absoluta, similarmente ao índice de coerência, é definido como o índice de dominância independente de ordem. Seu valor, D_a, é o limite superior para D e $0 \leq D_a \leq 1$.

$$D_a = \frac{\sum_{i<j}|r_{ij} - r_{ji}|}{m(m-1)/2}$$

Os índices D e D_a indicam a diferença média entre a credibilidade das declarações "a_i é preferível a a_j" e das declarações opostas "a_j é preferível a a_i" .

Note que C, I, D e C_a, I, D_a não são independentes umdo outro, ou seja:

$$C \cdot I = D \quad \text{e} \quad C_a \cdot I = D_a$$

d) ***Matriz normalizada.*** Uma matriz normalizada é obtida da matriz $\mathcal{R}$ usando a seguinte transformação:

$$r'_{ij} = \begin{cases} \frac{r_{ij}}{r_{ij} + r_{ji}} & \text{se } i \neq j \text{ e } r_{ij} + r_{ji} \neq 0 \\ r_{ij} & \text{demais casos.} \end{cases}$$

54.4 Método difuso-1: Camadas Não-dominadas

Os métodos de ordenamento baseados em lógica difusa assumem uma relação de preferências difusas com a função de filiação $\mu : \mathcal{A} \times \mathcal{A} \longrightarrow [0, 1]$ em um dado conjunto $\mathcal{A}$ de alternativas. Essa função de filiação é representada pela matriz $\mathcal{R}$ (ver seção 3 acima). Os valores $r_{ij} = \mu(a_i, a_j)$ são entendidos como os graus em que as preferências expressas pelas declarações "a_i é preferível a a_j" são verdadeiras.

Uma outra hipótese é a de que:

no caso de preferência fraca, μ é reflexiva, i.e.

$$\mu(a_i, a_i) = r_{ii} = 1 \qquad \text{para todo } a_i \in \mathcal{A}$$

no caso de preferencia estrita, μ e anti-reflexiva, i.e.

$$\mu(a_i, a_i) = r_{ii} = 0 \qquad \text{para todo } a_i \in \mathcal{A}$$

O procedimento de método difuso-1 procura por um CONJUNTO DE ALTERNATIVAS NÃO-DOMINADAS (denotado por alternativas ND), considerando tal conjunto como o núcleo de alternativas de maior nível. A razão para isso é que alternativas ND ou são equivalentes em relação a si mesmas, ou não são comparáveis tomando-se como base a relação de preferências considerada, e elas não são dominadas em um sentido estrito pelas outras alternativas.

Para determinar um conjunto difuso de alternativas ND, duas relações difusas correspondendo a relação de preferência $\mathcal{R}$ são definidas: relação de quasi-equivalência difusa e relação estrita difusa. Formalmente elas são definidas da seguinte maneira:

relação de quasi-equivalência $\mathcal{R}^e$:

$$\mathcal{R}^e = \mathcal{R} \cap \mathcal{R}^{-1}$$

relação de preferência estrita difusa $\mathcal{R}^s$:

$$\mathcal{R}^s = \mathcal{R} \setminus \mathcal{R}^e = \mathcal{R} \setminus (\mathcal{R} \cap \mathcal{R}^{-1}) = \mathcal{R} \setminus \mathcal{R}^{-1}$$

onde $\mathcal{R}^{-1}$ é uma relação oposta à relação $\mathcal{R}$.

Além do mais, a segunites funções de filiação são definidas respectivamente por $\mathcal{R}^e$ e $\mathcal{R}^s$:

$$\mu^e(a_i, a_j) = \min(r_{ij}, r_{ji})$$

$$\mu^s(a_i, a_j) = \begin{cases} r_{ij} - r_{ji} & \text{onde } r_{ij} > r_{ji} \\ 0 & \text{demais casos.} \end{cases}$$

Para qualquer alternativa fixa $a_j \in \mathcal{A}$, a função $\mu^s(a_j, a_i)$ descreve um conjunto difuso de alternativas que são estritamente dominadas por a_j. O complemento desse conjunto difuso, descrito pela função de filiação $1 - \mu^s(a_j, a_i)$, é para qualquer a_j fixo o conjunto difuso de todas as alternativas que não são estritamente dominadas por a_j. Então, a interseção de todos esses conjuntos difusos complementares (ao longo de todos os $a_j \in \mathcal{A}$) representa o conjunto difuso daquelas alternativas $a_i \in \mathcal{A}$ que não são estritamente dominadas por nenhuma das alternativas do conjunto $\mathcal{A}$. Esse conjunto é chamado de conjunto difuso μ^{ND} de alternativas ND no conjunto $\mathcal{A}$. Portanto, de acordo com a definição de interseção

$$\mu^{\text{ND}}(a_i) = \min_{a_j \in \mathcal{A}} (1 - \mu^s(a_j, a_i)) = 1 - \max_{a_j \in \mathcal{A}} \mu^s(a_j, a_i)$$

O valor $\mu^{\text{ND}}(a_i)$ representa o grau em que a alternativa a_i não é estritamente dominada por qualquer uma das alternativas do conjunto $\mathcal{A}$.

O NÚCLEO DE MAIOR NÍVEL DE ALTERNATIVAS contém aquelas alternativas a_i que oferecem o maior grau de não-dominância ou, em outras palavras, que fornece um valor de $\mu^{\text{ND}}(a_i)$ que é igual ao valor de:

$$M^{\text{ND}} = \max_{a_i \in \mathcal{A}} \mu^{\text{ND}}(a_i)$$

O valor de M^{ND} é chamado NÍVEL DE CERTEZA correspondendo ao núcleo definido por:

$$\mathcal{C}(\mathcal{A}) = \left\{ a_i \,|\, a_i \in \mathcal{A};\ \ \mu^{\text{ND}}(a_i) = M^{\text{ND}} \right\}$$

Os núcleos subseqüentes são construídos através da aplicação repetida do procedimento descrito acima. Os elementos do núcleo anterior são removidos da relação difusa primeiramente, i.e. as linhas e colunas correspondentes são removidas primeiramente da matriz de relação difusa. Entyão, os cálculos são repetidos na estrutura reduzida.

54.5 Método difuso-2: Ranges

A relação de entrada para esse método é a mesma do método-1, ou seja: a matriz $\mathcal{R}$ que deve ser reflexiva ou anti-reflexiva. No entanto, a questão a ser respondida aqui é bastante diferente.

O procedimento do método difuso-2 busca o NÍVEL DE CREDIBILIDADE, denotada por c_{jp}, DAS DECLARAÇÕES "a_j está exatamente na $p^{'esima}$ posição na seqüencia ordenada das alternativas em $\mathcal{A}$", denotada por T_{jp}. Os valores c_{jp} forma uma matriz $\mathcal{M}$ de $m \times m$ dimensões representando uma função de filiação difusa, cujas linhas correspondem às alternativas e as colunas às possíves posições na seqüência $1, 2, \ldots, m$.

Para tornar possível o cálculo dos c_{jp}'s, eles devem inicialmente ser decompostos em níveis de credibilidade já conhecidos r_{ij}, e, portanto, as declarações T_{jp} devem ser decompostos em declarações elementares com níveis de credibilidade conhecidos r_{ij}. Para esse fim, outra notação é introduzida. Note que para uma alternativa a_j estar exatamente no $p^{'esimo}$ lugar significa que é preferível a $m-p$ alternativas e é precedida pelas restantes $p-1$ alternativas. Quando o subconjunto de alternativas depois de a_j é fixado, então

$$\begin{array}{lcl} \mathcal{A}^j_{m-p} & = & \text{o subconjunto de alternativas onde } a_j \text{ e sempre preferivel,} \\ \mathcal{A}^j_{p-1} & = & \text{o subconjunto de alternativas que sao preferiveis a } a_j, \\ \mathcal{A}^j & = & \text{o subconjunto } \mathcal{A} \setminus \{a_j\}. \end{array}$$

Obviamente,

$$\mathcal{A}^j_{p-1} \cup \mathcal{A}^j_{m-p} = \mathcal{A}^j$$

$$\mathcal{A}^j_{p-1} \cap \mathcal{A}^j_{m-p} = \emptyset$$

e a declaração T_{jp} é equivalente a seqüência de declarações "a_j é preferível a todos os elementos de $\mathcal{A}^j_{m-p}$ e todos os elementos de $\mathcal{A}^j_{p-1}$ são preferíveis a a_j", conectados pelo operador disjuntivo lógico.

Além do mais, a declaração "a_j é preferível a todos os elementos de $\mathcal{A}^j_{m-p}$" é uma conjunção das já conhecidas declarações 'a_j é preferível a a_l", com o nível de credibilidade igual a r_{jl}, para todos os elementos a_l de $\mathcal{A}^j_{m-p}$.

Similarmente, a declaração "todos os elementos de $\mathcal{A}^j_{p-1}$ são preferíveis a a_j" é uma conjução das já conhecida declaração "a_i é preferível a a_j", com o nível de credibilidade igual a r_{ij}, para todos os elementos a_i de $\mathcal{A}^j_{m-p}$.

Aplicando os operadores difusos correspondentes, os elementos da matriz $\mathcal{M}$ podem ser obtidos da seguinte maneira:

$$c_{jp} = \max_{\mathcal{A}^j_{m-p} \subseteq \mathcal{A}^j} \left[\min \left(\min_{a_l \in \mathcal{A}^j_{m-p}} r_{jl}, \min_{a_i \in \mathcal{A}^j_{p-1}} r_{ij} \right) \right]$$

O cômputo dos valores de c_{jp} é executado usando-se um procedimento de otimização que produz uma série de subconjuntos $\mathcal{A}^j_{m-p}$ (enquanto se mantem j e p fixos) com valores monotonica e estritamente crescentes da função a ser maximizada nos passos sucessivos.

O programa fornece duas maneiras de interpretação da matriz $\mathcal{M}$.

CONJUNTOS DIFUSOS DE RANKS POR ALTERNATIVAS.

Para cada alternativa a_j, um valor de uma função de filiação difusa mostra a credibilidade de ter-se essa alternativa na $p^{'esima}$ posição ($p = 1, 2, \ldots, m$). Também, os ranks de maior credibilidade (posições) para cada alternativa são listados.

SUBCONJUNTOS DIFUSOS DE ALTERNATIVAS POR RANGES.

Para cada rank (posição) p, um valor de função de filiação difusa mostra a credibilidade da alternativa a_j ($j = 1, 2, \ldots, m$) estar nessa posição. Também as alternativas mais críveis, candidatas a uma posição, são listadas.

54.6 Referências

Dussaix, A.-M., *Deux méthodes de détermination de priorités ou de choix*, Partie 1: *Fondements mathématiques*, Document UNESCO/NS/ROU/624, UNESCO, Paris, 1984.

Jacquet-Lagrèze, E., Analyse d'opinions valuées et graphes de préférence, *Mathématiques et sciences humaines*, 33, 1971.

Jacquet-Lagrèze, E., L'agrégation des opinions individuelles, *Informatique et sciences humaines*, 4, 1969.

Kaufmann, A., *Introduction à la théorie des sous-ensembles flous*, Masson, Paris, 1975.

Orlovski, S.A., Decision-making with a fuzzy preference relation, *Fuzzy Sets and Systems*, Vol.1, No 3, 1978.

Capítulo 55

Diagramas de Dispersão

Notação

x	=	valor da variável a ser plotada horizontalmente
y	=	valor da variável a ser plotada verticalmente
w	=	valor de peso
k	=	subscrito para caso
N	=	número total de casos
W	=	soma total de pesos.

55.1 Estatísticas Univariadas

Essas estatísticas não-ponderadas são calculadas para todas variáveis usadas na execução.

a) ***Média.***

$$\overline{x} = \frac{\sum_k x_k}{N}$$

b) ***Desvio-padrão.***

$$s_x = \sqrt{\frac{\sum_k x_k^2}{N} - \overline{x}^2}$$

55.2 Estatísticas Univariadas Emparelhadas

Elas são calculadas no conjunto de casos possuindo dados válidos em ambos x e y. São estatísticas ponderadas, se uma variável de peso for especificada.

a) ***Média.***

$$\overline{x} = \frac{\sum_k w_k\, x_k}{W}$$

Note: a fórmula para $\overline{y}$ é análoga.

b) ***Desvio-padrão.***

$$s_x = \sqrt{\frac{\sum_k w_k\, x_k^2}{W} - \overline{x}^2}$$

Note: a fórmula para s_y é análoga.

c) ***N.*** O número de casos, ponderados, com dados válidos em x e y.

55.3 Estatísticas Bivariadas

Elas são calculadas no conjunto de casos que possuem dados válidos em ambos x e y.

a) ***Momento produto r de Pearson.***

$$r_{xy} = \frac{W \sum_k w_k\, x_k\, y_k - \left(\sum_k w_k\, x_k\right)\left(\sum_k w_k\, y_k\right)}{\sqrt{\left[W \sum_k w_k\, x_k^2 - \left(\sum_k w_k\, x_k\right)^2\right]\left[W \sum_k w_k\, y_k^2 - \left(\sum_k w_k\, y_k\right)^2\right]}}$$

b) ***Estatísticas de regressão: constante A e coeficiente B.***

$$A = \frac{\sum_k w_k\, y_k - \sum_k w_k\, x_k\, B}{W}$$

onde B é coeficiente de regressão não-padronizado.

$$B = \frac{W \sum_k w_k\, x_k\, y_k - \left(\sum_k w_k\, x_k\right)\left(\sum_k w_k\, y_k\right)}{W \sum_k w_k\, x_k^2 - \left(\sum_k w_k\, x_k\right)^2}$$

A constante A e o coeficiente B podem ser usados na equação de regressão $y = Bx + A$ para prever y de x.

Capítulo 56

Busca de Estrutura

Notação

y = valor da variável dependente
x = freqüência (ponderada) da variável dependente categórica
ou valores (ponderados) das variáveis dependentes dicotômicas
z = valor do covariate
w = valor do peso
k = subscrito para caso
j = subscrito para código de categoria da variável dependente
ou subscrito para variáveis dependentes dicotômicas
m = número de códigos da variável dependente
ou número de variáveis dependentes dicotômicas
g = subscrito para grupo; $g = 1$ indica a amostra completa
i = subscrito para grupos finais
t = número de grupos finais
N_g = número de casos no grupo g
W_g = soma dos pesos no grupo g
N_i = número de casos no grupo final i
W_i = soma dos pesos no grupo final i
N = número total de casos
W = soma total de pesos.

56.1 Análise de Médias

Esse método pode ser usado ao analisar-se uma variável dependente (intervalo ou dicotômica) e vários preditores. Seu objetivo é o de criar grupos que fornecerão a melhor predição dos valores da variável dependente a partir da média do grupo. Em outras palavras, grupos criados devem fornecer as maiores diferenças entre médias de grupos. Portanto, o critério de partição (variação explicada) é baseado em médias de grupos.

a) ***Estatísticas de traço.*** Essas são estatísticas calculadas na amostra completa (para $g = 1$), em tentativas de partições para grupos parentes, como também, para cada grupo resultando de uma partição ótima.

i) Soma (wt). Número de casos (N_g), se a variável de ponderação não for especificada, ou número ponderado de casos (W_g) no grupo g.

ii) MÉDIA Y. Valor médio da variável dependente y no grupo g.

$$\overline{y}_g = \frac{\sum_{k=1}^{N_g} w_k \, y_{gk}}{W_g}$$

iii) VAR Y. Variância da variável dependente y no grupo g.

$$\sigma^2_{y_g} = \frac{\sum_{k=1}^{N_g} w_k \, (y_{gk} - \overline{y}_g)^2}{W_g - \frac{W_g}{N_g}}$$

iv) VARIAÇÃO. Soma dos quadrados da variável dependente (como na análise de variância univariada) no grupo g.

$$V_g = \sum_{k=1}^{N_g} w_k \, (y_{gk} - \overline{y}_g)^2$$

v) VAR EXPL. Variação explicada é medida pela diferença entre a variação no grupo parente e a soma da variação nos dois grupos-criança. Ela fornece, para cada preditor, a quantidade de variação explicada pela melhor partição para esse preditor, i.e. o maior valor obtido em todas as partições para esse preditor.

Faça g_1 e g_2 denotar dois subgrupos (grupos-criança) obtidos em uma partição do grupo parente g, e V_{g_1} e V_{g_2}, suas respectivas variações. A variação explicada por tal partição do grupo g é calculada da seguinte maneira:

$$EV_g = V_g \; - \; (V_{g_1} \; + \; V_{g_2})$$

Então, esse valor é maximizado ao longo de todas as partições possíveis para o preditor.

vi) VARIAÇÃO EXPLICADA. Esse é o percentual da variação total explicado pelos grupos finais.

$$Percentual = 100 \, \frac{EV}{TV}$$

onde EV e TV são, respectivamente, a variação explicada pelos grupos finais e a variação total (ver 1.b abaixo).

b) ***Análise univariada de grupos finais.*** Essas são estatísticas de análise de variância univariada calculadas para os grupos finais.

i) VARIAÇÃO EXPLICADA e GL. Essa é a quantidade de variação explicada pelos grupos finais e os correspondentes graus de liberdade.

$$EV = TV - UV = TV - \sum_{i=1}^{t} V_i$$

$$GL = t - 1$$

ii) VARIAÇÃO TOTAL e GL. Variação calculada para a amostra completa, i.e. para grupo 1, e os graus de liberdade correspondentes.

$$TV = V_1$$

$$GL = W - 1$$

iii) ERRO e GL. Essa é a quantidade de variação não-explicada e os correspondentes graus de liberdade.

$$UV = \sum_{i=1}^{t} V_i$$

$$GL = W - t$$

c) ***Tabela de resumo de partição.*** A tabela fornece o valor de média de grupo, variância e variação da variável dependente a cada partição, como também a variação explicada pela partição (ver 1.a acima).

d) ***Tabela de resumo de grupos finais.*** A tabela fornece o valor de média de grupo, variância e variação da variável dependente para os grupos finais (ver 1.a acima).

e) ***Percentual de variância explicada.*** O percentual da variação total explicada pela melhor partição para cada grupo é calculado da seguinte maneira:

$$Percentual_g = 100\,\frac{EV_g}{TV}$$

Note que esse valor é igual a zero para os grupos finais (indicados por um asterisco).

f) ***Resíduos.*** Os resíduos são diferenças entre o valor observado e o valor predito da variável dependente.

$$e_k = y_k - \widehat{y}_k$$

Como valor predito, é designado a um caso o valor médio da variável dependente para o grupo onde ele pertence, i.e.

$$\widehat{y}_{ik} = \overline{y}_i$$

56.2 Análise de Regressão

Esse método pode ser usado ao se analisar uma variável dependente (intervalo ou dicotômica) com um covariate e vários preditores. Seu objetivo é de criar grupos que permitirão a melhor predição dos valores da variável dependente da equação de regressão de grupo e o valor do covariate. Em outras palavras, grupos criados devem fornecer as maiores diferenças nas linhas de regressão de grupo. O critério de partição (variação explicada) é baseado em regressão de grupos da variável dependente no covariate.

a) ***Estatísticas de traço.*** Essas são estatísticas calculadas em toda a amostra (para $g = 1$) e em partições tentativas para grupos parentes, como também, para cada grupo resultante da melhor partição.

i) Soma (wt). Número de casos (N_g) se a variável de ponderação não for especificada, ou número de casos ponderados (W_g) no grupo g.

ii) Média y,z. Valor médio da variável dependente y e do covariate z no grupo g (ver 1.a.ii acima).

iii) Var y,z. Variância da variável dependente y e o covariate z no grupo z (ver 1.a.iii acima).

iv) Inclinação. A inclinação da variável dependente y no covariate z no grupo g.

$$b_g = \frac{\sum_{k=1}^{N_g} w_k\,(y_{gk} - \overline{y}_g)(z_{gk} - \overline{z}_g)}{\sum_{k=1}^{N_g} w_k\,(z_{gk} - \overline{z}_g)^2}$$

v) Variação. Esse é o erro ou soma dos quadrados dos resíduos da estimação da variável dependente y pela sua regressão no covariate no grupo g, i.e. uma medida do desvio em relação a linha de regressão.

$$V_g = \sum_{k=1}^{N_g} w_k\,(y_{gk} - \overline{y}_g)^2 \;-\; b_g \times \sum_{k=1}^{N_g} w_k\,(y_{gk} - \overline{y}_g)(z_{gk} - \overline{z}_g)$$

onde b_g é a inclinação da linha de regressão no grupo g.

vi) Var expl. Variação explicada (EV). Ver 1.a.v acima para informações gerais, e 2.a.v acima para detalhes sobre V (variação) usados na análise de regressão.

vii) Variação explicada. É o percentual da variação total explicada pelos grupos finais. Ver 1.a.vi acima e 2.b abaixo.

b) ***Análise univariada de grupos finais.*** São as estatísticas resumo para os grupos finais. Ver 1.b acima para informações gerais, e 2.a.v e 2.a.vi acima para detalhes a cerca de medidas de V e EV usadas na análise de regressão.

c) ***Tabela de resumo de partição.*** A tabela fornece o valor da média de grupo, variância e variação da variável dependente a cada partição, como também a variação explicada por aquela partição. Ela também fornece o valor médio e variância do covariate. Ver 2.a acima para fórmulas. Além disso, as seguintes estatísticas de regressão são calculadas para cada partição:

i) INCLINAÇÃO. É a inclinação da variável dependente y no covariate z no grupo g (ver 2.a.iv acima).

ii) INTERCEPTO. É o termo constante da equação de regressão.

$$a_g = \overline{y}_g - b_g\,\overline{z}_g$$

onde b_g é a inclinação no grupo g.

iii) CORR. Coeficiente de correlação r de Pearson entre a variável dependente y e o covariate z no grupo g.

$$r_g = \frac{\sum_{k=1}^{N_g} w_k\,(y_{gk} - \overline{y}_g)\,(z_{gk} - \overline{z}_g)}{\sqrt{\sigma^2_{y_g}\,\sigma^2_{z_g}}}$$

d) ***Tabela de resumo de grupos finais.*** A tabela fornece a mesma informação (exceto a variação explicada) como na "Tabela de resumo de partição", mas para grupos finais.

e) ***Percentual de variação explicada.*** O percentual de variação total explicada pela melhor partição para cada grupo (ver 1.e e 2.a.vi acima).

f) ***Resíduos.*** Os resíduos são as diferenças entre o valor observado e o valor predito da variável dependente.

$$e_k = y_k - \widehat{y}_k$$

Valores preditos são calculados da seguinte maneira:

$$\widehat{y}_{ik} = a_i + b_i\,z_{ik}$$

onde a_i e b_i são coeficientes de regressão para o grupo final i.

56.3 Análise de Chi-quadrado

Esse método pode ser utilizado ao se analisar uma variável dependente (nominal ou ordinal) ou um conjunto de variáveis dependentes dicotômicas com vários preditores. Seu objetivo é o de criar grupos que permitirão o aparecimento da melhor predição da categoria da variável dependente a partir da sua distribuição de grupo. Em outras palavras, grupos criados devem fornecer as maiores diferenças nas distribuições das variáveis dependentes. O critério de partição (variação explicada) é calculado tomando como base as distribuições de freqüência da variável dependente. Note que variáveis dicotômicas dependentes múltiplas são tratadas como categorias de uma variável categórica.

a) ***Estatísticas de traço.*** Essas são as estatísticas calculadas na amostra completa (para $g = 1$), e em partições tentativas para grupos parentes, como também para cada grupo resultando de uma partição ótima.

i) SOMA (WT). Número de casos (N_g) se a variável de ponderação não for especificada, ou número de casos ponderados (W_g) no grupo g.

ii) VARIAÇÃO. Essa é a entropia para o grupo g, i.e. uma medida da desordem na distribuição da variável dependente.

$$V_g = -2 \sum_{j=1}^{m} x_{jg\cdot} \times \ln \frac{x_{jg\cdot}}{x_{\cdot g\cdot}}$$

onde

$$x_{jg\cdot} = \sum_{k=1}^{N_g} x_{jgk} \qquad x_{\cdot g\cdot} = \sum_{j=1}^{m} x_{jg\cdot}$$

e x_{jgk} é a "freqüência" (codificado 0 ou 1) do código j (ou valor da variável j) do caso k no grupo g.

iii) VAR EXPL. Variação explicada (EV). Ver 1.a.v acima para informações gerais, e 3.a.ii acima para detalhes sobre V (variação) usado na análise de chi-quadrado.

iv) VARIAÇÃO EXPLICADA. É o percentual da variação total explicada pelos grupos finais. Ver 1.a.vi acima e 3.b abaixo.

b) ***Análise univariada de grupos finais.*** São as estatísticas de resumo para os grupos finais. Ver 1.b acima para informações gerais, e 3.a.ii e 3.a.iii acima para detalhes sobre as medidas V e EV usadas na análise de chi-quadrado.

c) ***Tabela de resumo de partição.*** A tabela fornece a variação da variável dependente a cada partição, como também a variação explicada pela partição. Ver 3.a.ii e 3.a.iii acima para fórmulas.

d) ***Tabela de resumo de grupos finais.*** A tabela fornece a variação da variável dependente para os grupos finais.

e) ***Percentual de variância explicada.*** O percentual da variação total explicada pela melhor partição para cada grupo (ver 1.e e 3.a.iii acima).

f) ***Distribuições percentuais.*** Uma tabela bivariada mostrando as distribuições percentuais da variável dependente para todos os grupos (P_{jg}).

g) ***Resíduos.*** Os resíduos são as diferenças entre o valor observado e o valor predito da variável dependente.

Para análise com UMA VARIÁVEL DEPENDENTE CATEGÓRICA, resíduos são calculados para cada categoria da variável. Portanto, o número de resíduos é igual ao número de categorias.

$$e_{jk} = x_{jk} - \widehat{x}_{jik}$$

Valores observados, x_{jk}, são criados como uma série de "variáveis dummies", codificadas 0 ou 1.

Como valor predito para a categoria j, a proporção de casos pertencendo a categoria que o caso pertence é designado para um caso, i.e

$$\widehat{x}_{jik} = P_{ji}/100$$

Para análise com VÁRIAS VARIÁVEIS DEPENDENTES DICOTÔMICAS, resíduos são calculados para cada variável. Portanto, o número de resíduos é igual ao número de variáveis dependentes.

$$e_{jk} = x'_{jk} - \widehat{x}_{jik}$$

Valores observados são calculados da seguinte maneira:

$$x'_{jk} = \frac{x_{jk}}{\sum_{j=1}^{m} x_{jk}}$$

Como valor predito para a categoria j, a proporção de casos que possuem o valor de 1 para essa variável no grupo a qual pertencem é designado para um caso, i.e

$$\widehat{x}_{jik} = P_{ji}/100$$

56.4 Referências

Morgan, J.N., Messenger, R.C., *THAID A Sequential Analysis Program for the Analysis of Nominal Scale Dependent Variables*, Institute for Social Research, The University of Michigan, Ann Arbor, 1973.

Sonquist, J.A., Baker, E.L., Morgan, J.N., *Searching for Structure*, Revised ed., Institute for Social Research, The University of Michigan, Ann Arbor, 1974.

Capítulo 57

Tabelas Univariadas e Bivariadas

Notação

x	=	valor da variável de linha em tabelas bivariadas, ou valor da variável em tabelas univariadas
y	=	valor da variável de coluna em tabelas bivariadas
w	=	valor do peso
k	=	subscrito para caso
i	=	subscrito para linha em tabelas bivariadas
j	=	subscrito para coluna em tabelas bivariadas
r	=	número de linhas em tabelas bivariadas
c	=	número de colunas em tabelas bivariadas
$f_{i\cdot}$	=	freqüência marginal na linha i de uma tabela bivariada
$f_{\cdot j}$	=	freqüência marginal na coluna j de uma tabela bivariada
N	=	número total de casos.

57.1 Estatísticas Univariadas

a) ***Wtnum.*** O número da variável de ponderação ou zero, se a variável de ponderação não é especificada.

b) ***Wtsum.*** Número de casos, se a variável de ponderação não é especificada, ou o número ponderado de casos (soma dos pesos).

c) ***Moda.*** A primeira categoria que contiver a freqüência máxima.

d) ***Mediana.*** A mediana é calculada com n-quantis com dois subintervalos requisitados. Ver o capítulo "Funções de Distribuição e de Lorenz" para detalhes.

e) ***Média.***

$$\overline{x} = \frac{\sum_k w_k x_k}{\sum_k w_k}$$

f) ***Variância.*** Essa é uma estimativa não-viesada da variância da população.

$$\widehat{s}_x^2 = \left(\frac{N}{N-1}\right) \frac{\sum_k w_k \left(x_k - \overline{x}\right)^2}{\sum_k w_k}$$

g) ***Desvio-padrão.*** Deve-se notar que $\widehat{s}_x$ não é uma estimativa não-viesada do desvio-padrão da população.

$$\widehat{s}_x = \sqrt{\widehat{s}_x^2}$$

h) ***Coeficiente de variação (C.var.).***

$$C_x = \frac{100\ \widehat{s}_x}{\overline{x}}$$

i) ***Assimetria.*** A assimetria da distribuição de x é medida por

$$g_1 = \left(\frac{N}{N-2}\right)\left(\frac{m_3}{\widehat{s}_x^2\sqrt{\widehat{s}_x^2}}\right) \quad \text{onde} \quad m_3 = \frac{\sum\limits_k w_k\,(x_k - \overline{x})^3}{\sum\limits_k w_k}$$

Distribuições que são assimétricas a direita, i.e. a cauda está a direita, possuem assimetria positiva; distribuições que são assimétricas a esquerda possuem assimetria negativa; uma distribuição normal possui assimetria igual a 0.0.

j) ***Curtose.*** A curtose da distribuição de x é medida por

$$g_2 = \left(\frac{N}{N-3}\right)\left(\frac{m_4}{(\widehat{s}_x^2)^2}\right) - 3 \qquad \text{onde} \quad m_4 = \frac{\sum\limits_k w_k\,(x_k - \overline{x})^4}{\sum\limits_k w_k}$$

A curtose mede quão íngreme é a inclinação da distribuição em relação ao pico. Uma distribuição normal possui curtose igual a 0.0. Uma curva com uma inclinação maior possui curtose positiva; distribuições menos inclinadas do que a distribuição normal possuem uma curtose negativa.

k) ***n-quantis.*** Os pontos que definem os n-quantis são calculados do mesmo jeito que no programa QUANTILE.

57.2 Estatísticas Bivariadas

a) ***Chi-quadrado.*** Chi-quadrado é apropriado para testar a significância das diferenças entre distribuições de grupos independentes.

$$\chi^2 = \sum_i \sum_j \frac{(f_{ij} - E_{ij})^2}{E_{ij}}$$

onde

$$\begin{array}{rcl} f_{ij} & = & \text{a freqüência observada na célula } ij \\ E_{ij} & = & \text{a freqüência esperada(calculada) na célula } ij; \\ & & \text{é o produto da freqüência da linha } i \text{ vezes} \\ & & \text{a freqüência na coluna } j, \text{ dividido pelo total } N. \end{array}$$

Para tabelas dois por dois, a χ^2 é computada de acordo com a segunte fóromula:

$$\chi^2 = \frac{N(|ad - bc| - N/2)^2}{(a+b)(c+d)(a+c)(b+d)}$$

onde a, b, c, d representam as freqüências nas quatro células.

b) ***V de Cramer.*** O V de Cramer descreve a intensidade da associação na amostra. Seu valor vai de 0.0, refletindo completa independência, e 1.0, mostrando completa dependência dos atributos.

$$V = \sqrt{\frac{\chi^2}{N(L-1)}}$$

onde $L = \min(r, c)$.

c) ***Coeficiente de contingência.*** Como o V de Cramer, o coeficiente de contingência é usado para descrever a intensidade de associação na amostra. Seu limiter superior é uma função do número de categorias. O índice não pode alcaçar 1.0.

$$\text{CC} = \sqrt{\frac{\chi^2}{\chi^2 + N}}$$

d) ***Graus de liberdade.***

$$\text{gl} = (r-1)(c-1)$$

e) ***N ajustado.*** Esse é o N usado em computações estatísticas, i.e. o número de casos com códigos válidos. Será ponderado se uma variável de ponderação for especificada.

f) ***S.*** S é igual ao número de concordâncias na ordem menos o número de discordâncias na ordem. Para uma dada célula na tabela, todos os casos em células a direita e abaixo estão em concordância, todos os casos a esquerda e abaixo estão em discordância. S é o numerador da estatística tau e gama.

$$S = \sum_{i=1}^{r-1} \sum_{j=1}^{c} f_{ij} \left(\sum_{h=i+1}^{r} \sum_{l=j+1}^{c} f_{hl} - \sum_{m=i+1}^{r} \sum_{n=1}^{j-1} f_{mn} \right)$$

onde f_{ij}, f_{hl} e f_{mn} são as freqüências observadas nas células ij, hl e mn, respectivamente.

g) ***Variância de S.*** É a variância de S quando existir. (Um empate está presente nos dados se mais de um caso aparecer em uma dada linha ou coluna.)

$$\begin{aligned}
\sigma_s^2 &= \frac{N(N-1)(2N+5) - \sum_j f_{\cdot j}(f_{\cdot j}-1)(2f_{\cdot j}+5) - \sum_i f_{i\cdot}(f_{i\cdot}-1)(2f_{i\cdot}+5)}{18} + \\
&+ \frac{\left[\sum_j f_{\cdot j}(f_{\cdot j}-1)(f_{\cdot j}-2)\right]\left[\sum_i f_{i\cdot}(f_{i\cdot}-1)(f_{i\cdot}-2)\right]}{9N(N-1)(N-2)} + \\
&+ \frac{\left[\sum_j f_{\cdot j}(f_{\cdot j}-1)\right]\left[\sum_i f_{i\cdot}(f_{i\cdot}-1)\right]}{2N(N-1)}
\end{aligned}$$

h) ***Desvio-padrão de S.***

$$\sigma_s = \sqrt{\sigma_s^2}$$

i) ***Desvio normal de S.*** Fornece um teste de significância em grandes amostras para tau ou gama com empates. O -1 no numerador é uma correção para continuidade (se S é negativo, o 1 será adicionado). O valor pode ser consultado em uma tabela da distribuição normal. O teste é condicional à distribuição de empates.

$$Z = \frac{S-1}{\sigma_s}$$

j) ***Tau a.*** O τ de Kendall é uma medida da associação para dados ordinais. Tau a assume que não há empates nos dados, ou que empates, se presentes, representam uma "falha de medida" que está propriamente refletida por uma intensidade reduzida de relacionamento. Tau a pode variar de -1.0 a $+1.0$.

$$\tau_a = \frac{S}{\frac{N(N-1)}{2}}$$

k) ***Tau b.*** Tau b é como o tau a exceto que empates são permitidos, i.e. pode haver mais de um caso em uma dada linha ou coluna da tabela bivariada. Tau b pode assumir o valor de 1 somente quando o número de linhas for igual ao número de colunas.

$$\tau_b = \frac{S}{\sqrt{\left[\frac{N(N-1)}{2} - T_1\right]\left[\frac{N(N-1)}{2} - T_2\right]}}$$

onde

$$\begin{aligned} T_1 &= \left[\sum_i f_{i\cdot}(f_{i\cdot} - 1)\right] / 2 \\ T_2 &= \left[\sum_j f_{\cdot j}(f_{\cdot j} - 1)\right] / 2 \end{aligned}$$

l) ***Tau c.*** Tau c (também conhecido com tau de Kendall-Stuart) é como o tau b exceto que se o número de linhas não for igual ao número de colunas, tau b não pode assumir os valores $\pm$ 1.0, enquanto tau c assume esses valores.

$$\tau_c = \frac{S}{1/2\, N^2\, [(L-1)/L]}$$

onde $L = \min(r, c)$.

m) ***Gama.*** O γ de Goodman-Kruskal é outra medida de associação largamente utilizada que é proximamente relacionada ao tau de Kendall. Pode variar de -1.0 a $+1.0$ e pode ser computado mesmo que empates ocorram nos dados.

$$\gamma = \frac{S}{S_+ + S_-}$$

onde

$$\begin{aligned} S &= S_+ - S_- \\ S_+ &= \text{o número total de pares na mesma ordem} \\ S_- &= \text{o número total de pares em ordem diferente.} \end{aligned}$$

n) ***Ro de Spearman.*** Esse é o coeficiente de correlação produto-momento ordinário de Pearson calculado nos ranks. Varia de -1.0 a $+1.0$. O ro de Spearman computado por TABLES incorpora a correção para empates.

O fator de correção, T, para um grupo único de casos de empate é:

$$T = \frac{t^3 - t}{12}$$

onde t é igual ao número de casos empatados em um dado rank, i.e. o número de casos em uma dada linha ou coluna.

O ro de Spearman é calculado da seguinte maneira

$$\rho_s = \frac{\sum x^2 + \sum y^2 - \sum d^2}{2\sqrt{\sum x^2 \sum y^2}}$$

onde

$$\begin{aligned}
\sum x^2 &= \frac{N^3 - N}{12} - \sum T_x \\
\sum y^2 &= \frac{N^3 - N}{12} - \sum T_y \\
\sum d^2 &= \sum_k (X_k - Y_k)^2 \\
\sum T_x &= \text{a soma dos } T\text{'s para todas as linhas com mais de 1 caso} \\
\sum T_y &= \text{a soma dos } T\text{'s para todas as colunas com mais de 1 caso} \\
X_k &= \text{o rank do caso } k \text{ na variavel de linha} \\
Y_k &= \text{o rank do caso } k \text{ na variavel de coluna.}
\end{aligned}$$

Note que, quando mais de um caso ocorre em uma dada linha (ou coluna), o valor dos X_k's (ou Y_k's) para os casos empatados é a média dos ranks que seriam designados caso não tivessem ocorridos empates. Por exemplo, se há 15 casos na primeira linha da tabela, então seriam designa dos ranks para esses 15 casos, i.e. valor de X de 8.

o) ***Lambda simétrica.*** O lambda é uma medida simétrica da força de prever; é appropriada quando nem colunas nem linhas são designadas especialmente como a fonte da previsão, ou quando não são conhecidas. Lambda assume valores de 0 a 1.0 .

$$\lambda_{\text{sym}} = \frac{\sum_i \max_j f_{ij} + \sum_j \max_i f_{ij} - \max_j f_{\cdot j} - \max_i f_{i\cdot}}{2N - \max_j f_{\cdot j} - \max_i f_{i\cdot}}$$

onde

$$\begin{aligned}
f_{ij} &= \text{a freqüência observada na célula } ij \\
\max_j f_{ij} &= \text{a maior freqüência na linha } i \\
\max_i f_{ij} &= \text{a maior freqüência na coluna } j \\
\max_j f_{\cdot j} &= \text{a maior freqüência marginal dentre as colunas } j \\
\max_i f_{i\cdot} &= \text{a maior freqüência marginal dentre as linhas } i.
\end{aligned}$$

p) ***Lambda A, variável dependente na linha.*** Esse lambda é apropriado quando a variável de linha é a variável dependente. É uma medida da redução proporcional da probabilidade de erro ao prever-se a variável linha, que é calculada ao especificar-se a categoria da coluna. O lambda de linha dependente assume valores de 0 a 1.0 .

$$\lambda_{\text{rd}} = \frac{\sum_j \max_i f_{ij} - \max_i f_{i\cdot}}{N - \max_i f_{i\cdot}}$$

Ver acima as definições de termos nas fórmulas.

q) ***Lambda B, variável dependente na coluna.*** Esse lambda é apropriado quando a variável de coluna é a variável dependente. Ele assume os valores de 0 a 1.0.

$$\lambda_{\text{cd}} = \frac{\sum_i \max_j f_{ij} - \max_j f_{\cdot j}}{N - \max_j f_{\cdot j}}$$

Ver acima as definições de termos nas fórmulas.

r) ***Estatísticas de Medicina Baseada em Evidência (EBM).*** Elas são calculadas para tabelas 2x2 onde a primeira linha representa freqüências de evento (a) e não evento (b) para casos no grupo de tratados, e a segunda linha representa freqüências de evento (c) e não evento (d) no grupo de controle.

As seguintes estatísticas são calculadas:

Taxa de evento no grupo experimental

$$TEE = a/(a+b)$$

Taxa de evento no grupo de controle

$$TEC = c/(c+d)$$

Redução absoluta de risco (diferença de risco)

$$RAR = |TEC - TEE|$$

Redução relativa de risco

$$RRR = RAR/TEC$$

Número necessário para tratamento

$$NNT = 1/RAR$$

Risco relativo (quociente de risco)

$$RR = TEE/TEC$$

e seu intervalo de confiança de 95%

$$CI_{RR} = exp\left[\, ln(estimador\ RR) \pm 1.96\sqrt{T}\,\right]$$

onde a variância estimada de $ln(estimador\ RR)$ e

$$T = \frac{b/a}{a+b} + \frac{d/c}{c+d}$$

Razão de possibilidades (quociente de odds)

$$RP = ad/bc$$

e seu intervalo de confiança de 95%

$$CI_{RP} = exp\left[\, ln(estimador\ RP) \pm 1.96\sqrt{V}\,\right]$$

onde a variância estimada de $ln(estimador\ RP)$ e

$$V = \frac{1}{a} + \frac{1}{b} + \frac{1}{c} + \frac{1}{d}$$

s) ***Teste exato de Fisher.*** O teste de probabilidade exato de Fisher é uma técnica não-paramétrica extremamente útil para analisar dados discretos (nominais ou ordinais) oriundos de duas amostras independentes. É usado quando todos os casos de duas amostras aleatórias independentes pertencem a duas categorias mutuamente exclusivas. O teste determina se os dois grupos diferem na proporção com que eles estão presentes nas duas categorias.

Probabilidade do resultado observado é calculado da seguinte maneira:

$$p = \frac{(a+b)!\,(c+d)!\,(a+c)!\,(b+d)!}{N!\,a!\,b!\,c!\,d!}$$

onde a, b, c, d representam as freqüências nas quatro células.

O programa TABLES fornece as probabilidades exatas mono e bi-caudais, chamadas "probabilidade de resultado igual a ou mais extremo do que observado" e "probabilidade de resultado tão extremo quanto o observado em qualquer direção", respectivamente.

t) ***Teste de Mann-Whitney.*** O teste U de Mann-Whitney pode ser usado para testar se dois grupos independentes foram retirados da mesma população. É a alternativa mais útil em relação ao teste-t paramétrico quando a medida é mais fraca que o escalonamento do intervalo. No program TABLES é requerido que a variável linha seja a variável de agrupamento dicotômica.

Sejam

$$\begin{aligned} n_1 &= \text{o número de casos no menor dos dois grupos} \\ n_2 &= \text{o número de casos no segundo grupo} \\ R_1 &= \text{soma dos ranks designados para grupo com } n_1 \text{ casos} \\ R_2 &= \text{soma dos ranks designados para grupo com } n_2 \text{ casos.} \end{aligned}$$

Então

$$U_1 = n_1 n_2 + \frac{n_1(n_1+1)}{2} - R_1$$

$$U_2 = n_1 n_2 + \frac{n_2(n_2+1)}{2} - R_2$$

e

$$U = \min(U_1, U_2)$$

Se houver mais de 10 casos em cada grupo, o programa TABLES fornece aproximações Z (aproximações normais de U) calculadas da seguinte maneira:

$$Z = \frac{U - n_1 n_2/2}{\sqrt{\frac{n_1 n_2 (n_1 + n_2 + 1)}{12}}}$$

u) ***Teste "signed ranks" de Wilcoxon.*** O teste de Wilcoxon é um teste estatístico para duas amostras relacionadas e utiliza informação sobre ambas as direções e a magnitude relativa das diferenças entre os pares de variáveis.

A soma dos ranks positivos, T^+, é obtida da seguinte maneira:

- As diferenças com sinais $d_k = x_k - y_k$ são calculadas para todos os casos.
- As diferenças d_k são rankeadas sem se considerar seus sinais. Os casos com d_k's iguais a zero são eliminados. Será designado a média dos ranks empatados para os d_k's empatados.
- Em cada rank é afixado o sinal (+ ou −) do d que ele representa.
- N' é o número de d_k's não zero.
- T^+ é a soma dos d_k's positivos.

Se $N' > 15$, o programa computa a aproximação de Z (aproximação normal de T^+) da seguinte maneira:

$$Z = \frac{T^+ - \mu_{T^+}}{\sigma_{T^+}}$$

onde

$$\mu_{T^+} = \frac{N'(N'+1)}{4}$$

$$\sigma^2_{T^+} = \frac{N'(N'+1)\,(2N'+1)}{24} - \frac{1}{2}\sum_{t=1}^{g} n_t\,(n_t-1)\,(n_t-2)$$

e

$$\begin{aligned} g &= \text{o número de agrupamentos de diferentes ranks empatados} \\ n_t &= \text{o número de ranks empatados no agrupamento } t. \end{aligned}$$

Note que a aproximação Z é também ajustada para ranks empatados. O uso disso, no entanto, não produz mudança alguma na variância quando há empates.

v) ***Teste-t.*** Esse t-ratio é apropriado para testar a diferença entre duas médias independentes, i.e. duas amostras independentes. A variância é agupada.

$$t = \frac{\overline{y}_i - \overline{y}_h}{\sqrt{\left(\frac{n_i s_i^2 + n_h s_h^2}{n_i + n_h - 2}\right)\left(\frac{n_i + n_h}{n_i\, n_h}\right)}}$$

onde

$$\begin{aligned}
\overline{y}_i &= \text{a média da variável de coluna para casos na linha } i \\
\overline{y}_h &= \text{a média da variável de coluna para casos na linha } h \\
s_i^2 &= \text{a variância amostral da variável de coluna para casos na linha } i \\
s_h^2 &= \text{a variância amostral da variável de coluna para casos na linha } h.
\end{aligned}$$

Se testes-t são requisitados, desvios-padrões são calculados para os casos em cada linha da seguinte maneira:

$$s_i = \sqrt{\frac{\sum y^2}{n_i} - \overline{y}_i^2}$$

57.3 Nota sobre Pesos

Se estatísticas bivariadas são requisitadas e uma variável de ponderação é especificada, uma mensagem de precaução é impressa e a estatística é computada utilizando-se valores ponderados:

$$\begin{aligned}
x_k &= w_k\, x_k \\
x_k^2 &= w_k\, x_k^2 \\
y_k &= w_k\, y_k \\
y_k^2 &= w_k\, y_k^2 \\
N &= \sum_k w_k \\
f_{ij} &= \text{a freqüência ponderada na célula } ij.
\end{aligned}$$

Capítulo 58

Tipologia e Classificação Ascendente

Notação

x	=	valores de variáveis
k	=	subscrito para caso
v	=	subscrito para variável
g, i, j	=	subscrito para grupos
a	=	número de variáveis ativas (quantitativas e dicotomizadas qualitativas)
p	=	número de variáveis passivas (quantitativas e dicotomizadas qualitativas)
t	=	número de grupos iniciais
N_i	=	número de casos no grupo i (ponderados se o peso do caso for utilizado)
N_j	=	número de casos no grupo j (ponderados se o peso do caso for utilizado)
α	=	valor do peso de variável
w	=	valor do peso de caso
W	=	soma total dos pesos dos casos.

58.1 Tipos de Variáveis Utilizadas

O programa aceita tanto variáveis QUANTITATIVAS quanto QUALITATIVAS (categóricas), as últimas sendo tratadas como quantitativas depois de uma completa dicotomização das suas respectivas categorias, i.e. depois da construção de tantas variáveis dicotômicas (1/0) quanto o número de categorias. As variáveis usadas pelo programa podem ser ativas ou passivas. As VARIÁVEIS ATIVAS são aquelas em cuja base a tipologia é construída.

As VARIÁVEIS PASSIVAS não participam na construção da tipologia, mas o programa imprime para elas as principais estatísticas dentro dos grupos de tipologia.

Um conjunto de variáveis ativas é denotado aqui $\mathcal{X}_a$, e um conjunto de variáveis passivas $\mathcal{X}_p$.

58.2 Perfil de Caso

O perfil do caso k é um vetor $\mathcal{P}_k$ tal que

$$\mathcal{P}_k = (x_{k1}, x_{k2}, \ldots, x_{kv}, \ldots, x_{ka}) = (x_{kv})$$

onde todo $x_v \in \mathcal{X}_a$.

Se as variáveis ativas são requisitadas a serem padronizadas, o k^{esimo} perfil de caso se torna

$$\mathcal{P}_k = \left(\frac{x_{kv}}{s_v}\right)$$

onde s_v é o desvio-padrão da variável x_v (ver 7.b abaixo).

58.3 Perfil de Grupo

Perfil do grupo i, chamado também de baricentro do grupo, é um vetor $\mathcal{P}_i$ tal que

$$\mathcal{P}_i = (\overline{x}_{i1}, \overline{x}_{i2}, \ldots, \overline{x}_{iv}, \ldots, \overline{x}_{ia}) = (\overline{x}_{iv})$$

e no caso de dados padronizados, se torna

$$\mathcal{P}_i = \left(\frac{\overline{x}_{iv}}{s_v}\right)$$

onde o numerador é a média da variável x_v para os casos pertencendo ao grupo i e o denominador é o desvio-padrão geral dessa variável.

58.4 Distâncias Usadas

Há três tipos básicos de distâncias usadas no programa, ou seja: distância city-block, distância euclidiana e distância chi-quadrado de Benzécri. Elas podem ser utilizadas para calcular distâncias entre dois casos, entre um caso e um grupo de casos e entre dois grupos de casos. Abaixo, essas distâncias são definidas como distâncias entre dois grupos de casos (entre dois perfis de casos), mas as outras distâncias podem facilmente ser obtidas ao adaptar-se as respectivas fórmulas.

a) ***Distância city-block.***

$$d_{ij} = d(\mathcal{P}_i, \mathcal{P}_j) = \frac{\sum_{v=1}^{a} \alpha_v \, |\overline{x}_{iv} - \overline{x}_{jv}|}{\sum_{v=1}^{a} \alpha_v}$$

b) ***Distância euclidiana.***

$$d_{ij} = d(\mathcal{P}_i, \mathcal{P}_j) = \sqrt{\frac{\sum_{v=1}^{a} \alpha_v \, (\overline{x}_{iv} - \overline{x}_{jv})^2}{\sum_{v=1}^{a} \alpha_v}}$$

c) ***Distância chi-quadrado.***

$$d_{ij} = d(\mathcal{P}_i, \mathcal{P}_j) = \sqrt{\sum_{v=1}^{a} \frac{1}{p_v} \left(\frac{p_{iv}}{p_i} - \frac{p_{jv}}{p_j}\right)^2}$$

onde

$$p_v = \sum_{g=1}^{t} \overline{x}_{gv} \ , \quad p_i = \sum_{v=1}^{a} \overline{x}_{iv} \ , \quad p_j = \sum_{v=1}^{a} \overline{x}_{jv}$$

$$p_{iv} = \frac{\overline{x}_{iv}}{\sum_{g=1}^{t} \sum_{v=1}^{a} \overline{x}_{gv}} \ , \quad p_{jv} = \frac{\overline{x}_{jv}}{\sum_{g=1}^{t} \sum_{v=1}^{a} \overline{x}_{gv}}$$

Além disso, o programa propicia a possibilidade de usar-se distância "ponderada", chamada DESLOCAMENTO, que é definida da seguinte maneira:

$$D_{ij} = D(\mathcal{P}_i, \mathcal{P}_j) = \frac{2N_iN_j}{N_i + N_j}\, d_{ij}$$

Note que deslocamento entre dois perfis de casos é igual às suas distâncias, pois $N_i = N_j = 1$.

58.5 Construindo uma Tipologia Inicial

a) ***Seleção de uma configuração inicial.*** Antes de começar o processo de agregação de casos, o programa seleciona a configuração inicial, i.e. t perfis de grupos iniciais, de acordo com uma das seguintes maneiras:

- perfis de casos de t casos aleatoriamentes selecionados (usando números aleatórios) constituem a configuração inicial; para obter-se a configuração inicial, os casos remanescentes são distribuídos em t grupos como descrito abaixo;
- perfis de casos de t casos selecionados de uma maneira por passos constituem a configuração inicial; para obter a configuração inicial, os seguintes casos são distribuídos em t grupos como descrito abaixo;
- a configuração inicial em um conjunto de perfis de grupo calculados para casos distribuídos ao longo de categorias de uma variável chave;
- a configuração inicial é um conjunto de perfis de grupos "a priori" fornecidos pelo usuário.

Quando a construção começa de t perfis de casos, o programa considera esse conjunto de t vetores como um conjunto de t "casos iniciais" e distribui os casos remanescentes de acordo com sua distância entre si e o caso inicial.

Vamos chamar o conjunto de t casos iniciais de

$$\mathcal{P}_{\text{inicial}} = \Big\{\mathcal{P}_{k_1}, \mathcal{P}_{k_2}, \ldots, \mathcal{P}_{k_t}\Big\}$$

e a distância entre grupos e/ou casos i e j por $D(\mathcal{P}_i, \mathcal{P}_j)$.

Note que $D(\mathcal{P}_i, \mathcal{P}_j)$ pode ser qualquer distância definida na seção 4 acima.

Para cada caso $i \notin \mathcal{P}_{\text{inicial}}$ o programa calcula

$$\beta = \min_{1 \le j \le t} \Big[D(\mathcal{P}_i, \mathcal{P}_{k_j})\Big]$$

$$\gamma = \min\Big[D(\mathcal{P}_{k_1}, \mathcal{P}_{k_2}), D(\mathcal{P}_{k_1}, \mathcal{P}_{k_3}), \ldots, D(\mathcal{P}_{k_{t-1}}, \mathcal{P}_{k_t})\Big]$$

Existem duas possibilidades:

- $\beta \le \gamma$: caso i é designado para o grupo mais próximo $\mathcal{P}_{k_j}$ e o perfil desse grupo é recalculado

  $$\mathcal{P}_{k_j} = \Big(\mathcal{P}_{k_j} + \mathcal{P}_i\Big)/2$$

- $\beta > \gamma$: caso i forma um novo grupo que é adicionado ao conjunto $\mathcal{P}_{\text{inicial}}$, e os dois perfis mais próximos $\mathcal{P}_{k_j}$ and $\mathcal{P}_{k_{j'}}$ são agregados formando um grupo com o novo perfil

  $$\mathcal{P}_{k_j} = \Big(\mathcal{P}_{k_j} + \mathcal{P}_{k_{j'}}\Big)/2$$

No final desse procedimento, a configuração inicial é um conjunto de t perfis

$$\mathcal{P}_{\text{initial}} = \Big\{\mathcal{P}_1, \mathcal{P}_2, \ldots, \mathcal{P}_j, \ldots, \mathcal{P}_t\Big\}$$

onde $\mathcal{P}_j$ é um perfil médio de todos os casos pertencendo ao grupo j.

Nesse estágio, o programa não leva em consideração a ponderação de casos.

b) ***Estabilização da configuração inicial.*** A configuração inicial é estabilizada por um processo iterativo. Durante cada iteração, o programa redistribui os casos entre os grupos iniciais considerando suas distâncias em relação a cada perfil de grupo.

Aqui novamente existem duas possibilidades:

- quando caso $i \in \mathcal{P}_j$ e

$$D(\mathcal{P}_i, \mathcal{P}_j) = \min_{1 \le g \le t} \Big[D(\mathcal{P}_i, \mathcal{P}_g) \Big]$$

então, esse caso mantém-se no grupo $\mathcal{P}_j$;

- quando caso $i \in \mathcal{P}_j$ mas

$$D(\mathcal{P}_i, \mathcal{P}_{j'}) = \min_{1 \le g \le t} \Big[D(\mathcal{P}_i, \mathcal{P}_g) \Big]$$

então o caso i é movido do grupo $\mathcal{P}_j$ para o grupo $\mathcal{P}_{j'}$, e os perfis desses dois grupos são recalculados da seguinte maneira:

$$\mathcal{P}_j = \left(N_j \mathcal{P}_j - \mathcal{P}_i \right) / (N_j - 1)$$

$$\mathcal{P}_{j'} = \left(N_{j'} \mathcal{P}_{j'} + \mathcal{P}_i \right) / (N_{j'} + 1)$$

Depois dessa operação, o grupo $\mathcal{P}_j$ contém $N_j - 1$ casos e o grupo $\mathcal{P}_{j'}$ contém $N_{j'} + 1$ casos.

Note que, se os casos são ponderados, então

$$N_j = N_j - w_i$$

$$N_{j'} = N_{j'} + w_i$$

$$\mathcal{P}_i = w_i \, \mathcal{P}_i$$

onde w_i é o peso do caso i, e N_j e $N_{j'}$ são os números ponderados de casos nos grupos $\mathcal{P}_j$ e $\mathcal{P}_{j'}$, respectivamente.

Estabilidade de grupos é medida pela percentagem de casos que não mudam de grupo entre duas iterações subseqüentes.

O procedimento é repetido até que os grupos estejam estabilizados ou quando o número de iterações fixadas pelo usuário seja alcançado.

58.6 Características de Distâncias por Grupos

a) ***N.*** O número de casos em cada grupo da tipologia inicial.

b) ***Média.*** Distância média para cada grupo, i.e. a média de distâncias do perfil do grupo em relação a todos os casos pertencendo a esse grupo.

c) ***DP.*** Desvio-padrão da distância de cada grupo.

d) ***Classificação de distâncias.*** Distribuição de casos, tanto em termos de freqüências quanto percentagens, ao longo de 15 intervalos contínuos, que são diferentes para cada grupo.

e) ***Contagem total.*** Número total de casos participando na construção da tipologia inicial.

f) ***Média.*** Distância média geral.

g) ***DP.*** Desvio-padrão geral.

h) ***Classificação de distâncias (mesmos limites para cada grupo).*** Mesmo que 6.d acima, exceto que os 15 intervalos posuem a mesma amplitude para todos os grupos.

58.7 Estatísticas de Resumo para Variáveis Quantitativas e para Variáveis Qualitativas Ativas

a) ***Média.*** Média de quantitativas $x_v \in (\mathcal{X}_a \cup \mathcal{X}_p)$. Para categorias de variável qualitativa, é uma proporção de casos nessa categoria.

$$\overline{x}_v = \frac{\sum_k w_k \, x_{kv}}{W}$$

b) ***D.P.*** Desvio-padrão.

$$s_v = \sqrt{\frac{W \sum_k w_k \, x_{kv}^2 - \left(\sum_k w_k \, x_{kv}\right)^2}{W^2}}$$

c) ***Peso.*** O valor do peso da variável calculado para cada variável da seguinte maneira:

$$\alpha_v = \begin{cases} 0 & \text{para variáveis passivas quantitativas} \\ 1 & \text{para variáveis ativas quantitativas} \\ \frac{\sqrt{(c+1)/3}}{c} & \text{para categorias de uma variável ativa qualitativa,} \\ & \text{onde } c \text{ é o número de categorias não-vazias} \\ & \text{da variável sob consideração} \\ 1 & \text{para categorias de uma variável ativa qualitativa} \\ & \text{se distância chi-quadrado é utilizada.} \end{cases}$$

58.8 Descrição de Tipologia Resultante

No final da construção da tipologia inicial e no final de cada passo de classificação ascendente, todas as variáveis, i.e. ativas e passivas, são avaliadas pela quantidade de variância explicada. É uma medida de potência discriminante de cada variável quantitativa e cada categoria de variáveis qualitativas. Isso é seguido por uma descrição individual de todos os grupos da tipologia.

a) ***Proporção de casos.*** Percentual, multiplicado por 1000, de casos pertencendo a cada grupo da tipologia.

b) ***Variância explicada.***

$$\mathrm{EV}(x_v) = \frac{\sum_{i=1}^{t_g} N_i \left(\overline{x}_{iv} - \overline{x}_v\right)^2}{\sum_k w_k \left(x_{kv} - \overline{x}_v\right)^2} \times 1000$$

onde

$$\begin{aligned} t_g &= \text{número de grupos na tipologia} \\ \overline{x}_{iv} &= \text{média da variável } v \text{ no grupo } i \\ \overline{x}_v &= \text{média geral da variável } v. \end{aligned}$$

c) ***Média geral.***

Para variáveis QUANTITATIVAS, valores médios como descrito em 7.a acima.

Para CADA CATEGORIA DE variáveis QUALITATIVAS, percentual de casos nessa categoria.

d) ***Estatísticas para cada grupo da tipologia.***

Para variáveis QUANTITATIVAS:
primeira linha: valores médios descritos em 7.a acima;
segunda linha: desvios-padrões como descritos em 7.b acima.
Para CADA CATEGORIA DE variáveis QUALITATIVAS:
primeira linha: percentual de coluna de casos;
segunda linha: percentual de linha de casos.

58.9 Resumo da Quantidade de Variância Explicada pela Tipologia

Similarmente à descrição da tipologia resultante, uma tabela sumário é impressa no final da construção da tipologia inicial e ao final de cada passo de classificação ascendente.

a) ***Variáveis explicando*** 80% ***da variância.*** Lista das variáveis mais discriminantes, i.e. aquelas variáveis – tomadas juntas – responsáveis por pelo menos 80% da variância explicada, juntamente com a variância explicada por cada um deles individualmente (ver 8.b acima).

b) ***Variância média explicada pelas variáveis ativas.***

$$\overline{\mathrm{EV}}_{\mathrm{ativas}} = \frac{\sum_{v=1}^{a} \alpha_v \, \mathrm{EV}(x_v)}{\sum_{v=1}^{a} \alpha_v}$$

c) ***Variância média explicada por todas variáveis.***

$$\overline{\mathrm{EV}}_{\mathrm{todas}} = \frac{\sum_{v=1}^{a+p} \alpha_v \, \mathrm{EV}(x_v)}{\sum_{v=1}^{a+p} \alpha_v}$$

d) ***Variância média explicada pelas variáveis que explicam*** 80% ***da variância total.*** Após cada reagrupamento, o programa procura por variáveis que explicam pelo menos 80% da variância total (ver 9.a acima) e imprime a variância média explicada por aquelas variáveis antes e depois do reagrupamento, e o percentual de tais variáveis.

58.10 Classificação Ascendente Hierárquica

Depois da criação da tipologia inicial, o programa executa uma seqüência de reagrupamentos, reduzindo um por um o número de grupos até o número especificado pelo usuário. A cada reagrupamento, o programa seleciona dois grupos mais próximos, i.e. dois grupos com a menor distância ou deslocamento (ver seção 4 acima), e calcula o perfil para esse novo grupo.

a) ***Grupo i + j.*** Perfil do novo grupo, impresso para até 15 variáveis ativas em ordem decrescente do seus desvios (ver 10.d abaixo). Note que se houver menos do que 15 variáveis ativas, ou menos do que 15 variáveis com casos válidos em grupos agregados, o programa completa a lista usando variáveis passivas.

b) ***Grupo i.*** Perfil do grupo i, impresso para as mesmas variáveis como acima.

c) ***Grupo j.*** Perfil do grupo j, impresso para as mesmas variáveis como acima.

d) ***Desv.*** Valor absoluto da diferença entre os perfis do grupos i e j, impresso para as mesmas variáveis como acima.

$$\text{Dev}(x_v) = |\overline{x}_{iv} - \overline{x}_{jv}|$$

e) ***Desvio ponderado.*** Desvio ponderado pelo peso e desvio-padrão de variável, impresso para a mesma variável com acima.

$$\text{WDev}(x_v) = \text{Dev}(x_v)\ \frac{\alpha_v}{s_v}$$

58.11 Referências

Aimetti, J.P., *SYSTIT: Programme de classification automatique*, GSIE-CFRO, Paris, 1978.

Diday, E., *Optimisation en classification automatique*, RAIRO, Vol. 3, 1972.

Hall & Ball, A clustering technique for summarizing multivariate data, *Behavioral Sciences*, Vol. 12, No 2, 1967.

Apêndice

Mensagens de Erro dos Programas do IDAMS

Visão Geral

Um esforço foi feito no sentido de tornar as mensagens de erro auto-explicativas. Portanto, esse Apêndice essencialmente descreve o esquema de codificação usado para as mensagens de erro.

Erros e Alertas

Erros (E) causam sempre a interrupção da execução do programa do IDAMS, enquanto que alertas (W) alertam os usuários a respeito de possíveis anormalidades nos dados e/ou nas declarações de controle, e também sobre possíveis interpretações errôneas dos resultados. Mensagens de erro e de alerta possuem o seguinte formato:

`***E* aaannn` texto de mensagem de erro

`***W* aaannn` texto de mensagem de alerta

onde

`nnn` é um número de três dígitos, começando de 001 para alertas e de 101 para erros;

`aaa` indica de onde vem a mensagem, de acordo com as seguintes regras:

- ***Mensagens de programas:*** a primeira letra do nome do programa seguida pelas duas próximas consoantes no nome do programa.
- ***Mensagens de subrotinas:***

  SYN erros de sintaxe em geral;

  RCD erros e alertas de Recode (sintaxe);

  DTM erros de dados e dicionário, e alertas sobre arquivos de dados e dicionário;

  SYS erros e alertas do Monitor;

  FLM erros e alertas de administração de arquivos.

Mensagens de Erro de Execução do Fortran

Quando ocorrem erros durante a execução do programa (run time), o Visual Fortran RTL libera uma mensagens de diagnóstico. Elas possuem o segunte formato:

forrtl: severity (number): text

forrtl	Identifica a fonte como de Visual Fortran RTL.
severity	Os níveis de severidade são: severo (deve ser corrigido), erro (deveria ser corrigido), alerta (deveria ser investigado), ou info (para finalidade apenas de informar).
number	É o número da mensagen, e também o valor IOSTAT para declarações I/O.
text	Explica o evento que causou a mensagem.

As mensagens de erro de execução são auto-explicativas e portanto não estão listadas aqui.

Índice

agregação de dados, 45, 50, 97
agrupamento hierárquico
  aglomerativo, 174, 341
  baseado em variáveis dicotômicas, 174, 343
  divisivo, 174, 342
análise
  de agrupamento, 173, 337
  de classificação múltipla, 223
  de correlação, 255, 397
  de correspondências, 195
  de escalonamento, 215, 371
  de ordenamento, 261, 401
    lógica clássica, 261, 402
    lógica difusa, 261, 406, 408
  de regressão, 205
  de séries temporais, 329, 333
  de scoring, 247, 393
  de segmentação, 275, 413
  de variância, 223, 241, 377, 389
  discriminante, 185, 349
  espacial, 179, 345
  fatorial, 195, 357
    de componentes principais, 195
    discriminante, 186, 351
análise de variância
  multivariada, 233
arquivos
  classificando, 157
  Dados, 79
  de sistema, 80
    permanente, 80
    temporários, 80
  de usuário, 79
  Dicionário, 79
  especificando no IDAMS, 22
  fundindo, 157
  limitações de tamanho para o IDAMS, 12
  Matriz, 79
  Resultados, 79
  Setup, 79
  usados no WinIDAMS, 79
assimetria, 358, 420
auto-correlação, 333
autoregressão, 333

bivariadas
  estatísticas, 283, 308, 420
    produzidas por TABLES, 286
  tabelas, 283, 307
    apresentação gráfica, 308
    produzidas por TABLES, 286

código
  checagem, 58, 109
  labels, 16
código de condição
  ajustando para erros de declarações de controle, 21
  checando entre programas, 21
campos em branco, 13
casas decimais, especificação, 15
caso
  ativo, 195, 362
  criando vários casos a partir de um, 49
  deleção, 129, 161
  especificando número de registros por caso, 14
  identificação (ID)
    correção, 129
  limitações de tamanho, 12
  listagem, 145, 165
  listando, 129
  passivo, 195, 364
  principal, 195, 362
  seleção
    com filtro, 25
    com Recode, 49
  suplementar, 195, 364
checando
  códigos, 58, 109
  consistência, 59, 115
  estrutura de dados, 58, 121
  intervalo de valores, 58
  ordem de casos, 131
  ordem de classificação, 161
  range de valores, 109
chi-quadrado
  distância, 299, 428
  teste, 283, 308, 420
classificação de objetos
  baseada em agrupamento hierárquico, 174, 341–343
  baseada em lógica difusa, 174, 340
  baseada em partição, 174, 338, 340
classificando
  arquivos, 157
  dados, 89
coeficientes
  B, 207, 256, 269, 368, 398, 412
  beta, 207, 225, 368, 379
  de correlação múltipla, 207, 367

de correlação parcial, 207, 366
de Gini, 191, 354
de variação, 365, 377, 378, 389, 420
eta, 225, 242, 379, 390
r de Pearson, 255, 397
termo constante, 207, 256, 269, 368, 398, 412
comandos do IDAMS, 21
$CHECK, 21
$COMMENT, 22
$DATA, 22
$DICT, 22
$FILES, 22
$MATRIX, 22
$PRINT, 22
$RECODE, 22
$RUN, 23
$SETUP, 23
combinando datasets
a níveis diferentes, 149
em um mesmo nível, 149
comentários no setup do IDAMS, 22
configuração
análise, 179, 345
centragem, 345, 371
matriz, 345, 371, 374
entrada em CONFIG, 180
entrada em TYPOL, 298
produzida por CONFIG, 180
produzida por MDSCAL, 217
produzida por TYPOL, 297
normalização, 345, 371
projeção, 180
rotação, 179, 345
rotação varimax, 180, 346
transformação, 179, 346
contingência
coeficiente, 283, 308, 421
tabelas, 283
copiando datasets, 161
correlação
coeficientes, 255, 397
matriz, 398
entrada em CLUSFIND, 175
entrada em MDSCAL, 217
entrada em REGRESSN, 208
produzida por PEARSON, 256
produzida por REGRESSN, 206, 207
parcial, 207, 366
corrigindo
dados, 59, 88, 129
ID de caso, 129
variáveis, 129
covariâncias
matriz
produzida por PEARSON, 257
Cramer (V de), 283, 421
critério de Kaiser, 199
curtose, 358, 420

D de Sormer, 308
dados
agregação, 97
checagem de estrutura, 58, 121
classificando, 89
correção, 59, 88, 129
edição, 14, 57, 103
entrada, 88
exportação
em formato DIF, 136
em formato livre, 90, 137
formato em IDAMS, 12
importação, 20
em formato DIF, 137
em formato livre, 89, 137
listagem, 145
no setup de entrada, 22
recodificação, 59
transformação, 59, 165
validação, 57, 109, 115, 121
dados perdidos
a serem usados para checagem, 30
códigos
designados por Recode, 51
especificação, 13, 15
checando com Recode, 45
definição, 13
deleção por casos
em Pearson, 255
em REGRESSN, 206
deleção por pares
em Pearson, 255
manuseio via Recode, 34
dataset
cópia, 161
construção, 103
criação de subconjuntos, 161
definição no IDAMS, 11
intercalação, 149
ddname, 23
para arquivos de dicionário e de dados, 30
decis, 191, 285, 353, 420
declarações de controle, 25
filtro, 25
parâmetros, 27
regras de codificação, 25
título, 27
default nos parâmetros do IDAMS, 27
deletando
casos, 129, 161, 165
variáveis, 161, 165
densidades, 322
desvio-padrão, 349, 357, 365, 377, 378, 389, 397, 398, 411, 412, 420, 431
diagramas de dispersão, 269
diagrama agrupado, 324
manipulação, 320
rotação, 325
tridimensional, 324
dicionário, 14

cópia, 161
criação, 86, 103
descrevedor de variável (registro-T), 15
exemplo, 16
label de código (registros-C), 15
listagem, 145
no setup de entrada, 22
registro de descrição, 14
verificação, 87
discriminante
análise, 185, 349
análise fatorial, 186, 351
função, 185, 350
distância
chi-quadrado, 299, 428
city-block, 176, 219, 299, 338, 375, 428
de Mahalanobis, 185, 350
euclidiana, 176, 215, 219, 299, 338, 374, 428
distribuições
de freqüências, 283, 305
marginais, 283
duplicados
casos, deleção, 161, 163
registros, detecção e deleção, 122
Durbin-Watson (teste), 207, 369

editando
arquivos de texto, 93
dados, 57
valores de dados não-numéricos, 30, 103
escalonamento multidimensional, 215, 371
escores
calculados por FACTOR, 197, 363, 364
calculados por POSCOR, 248, 395
espaços em branco
detecção, 112
recodificação, 30, 103
especificação de subconjunto
em POSCOR, 251
em QUANTILE, 193
em TABLES, 288
espectro, 334
espectro cruzado, 334
estatísticas
bivariadas, 283
de EBM, 283
descritivas, 97, 98, 196, 269, 283, 305, 306, 357, 411, 419
EBM, 424
gama, 283, 422
lambda, 283, 308, 423
ro de Spearman, 283, 422
tau, 283, 308, 422
univariadas, 97, 98, 207, 283
estimação de tendência, 333
exploração gráfica de dados, 317
exportação
de dados, 135
de datasets, 6
de matrizes, 6, 135
de tabelas multidimensionais, 308

fator de repetição
em TABLES, 288
filtro
com variáveis-R, 49
declaração de controle, 25
local
em ONEWAY, 244
em QUANTILE, 194
em SCAT, 272
em TABLES, 288
localização, 25
regras de codificação, 25
verificação de sintaxe, 92
filtros de freqüência, 334
Fisher
teste exato, 283, 424
teste F, 207, 225, 242, 367, 391
folders
default, 80
usados em WinIDAMS, 80
função
de distribuição, 191, 353
de Lorenz, 191, 354
discriminante, 185, 350
fundindo arquivos, 157

gama (estatística), 283, 308, 422
Gini (coeficiente), 191, 354
gráficos de box-whisker, 323

histogramas, 322, 333

IDAMS
dataset, 11
construção, 103
declarações de controle, 25
dicionário, 14
execução de programas, 92
manuseio de resultados, 92
matriz, 16
exportação, 135
importação, 135
mensagens de erro, 435
setup, 21
preparação, 91
verificação, 92
importação
de dados, 89, 90, 135
de datasets, 6
de matrizes, 6, 135
imprimindo o setup do IDAMS, 22
interação
definição, 223
detecção e tratamento, 223
intercalando datasets, 149

Kaiser (critério de), 199
Kendall (taus), 283, 422
Kolmogorov-Smirnov (teste D), 191, 194, 354

lógica difusa
  classificação de objetos, 174, 340
  ordenamento de alternativas, 261, 406, 408
label para categorias de código, 16
lambda (estatística), 283, 423
linha de continuação
  declarações de controle, 25
  declarações de Recode, 33
lista de variável
  regras de codificação, 31
listando
  casos, 129, 145
  dados, 145, 165
  dicionário, 145
Lorenz
  curva, 354
  função, 191, 354

média, 337, 349, 357, 365, 377, 378, 383, 389, 397, 398, 411, 419, 431
método de ordenamento ELECTRE, 261
métrica-r de Minkowski, 215, 374
Mahalanobis (distância), 185, 350
Mann-Whitney (teste), 283, 425
matriz
  de covariâncias, 398
  de configuração
    entrada em MDSCAL, 218
  de correlações, 359, 366, 398
    entrada em CLUSFIND, 175
    entrada em MDSCAL, 217
    entrada em REGRESSN, 208
    parciais, 207, 366
    produzida por PEARSON, 256
    produzida por REGRESSN, 206, 207
  de covariâncias, 359
    produzida por PEARSON, 257
  de dissimilaridades, 173, 338
    entrada em CLUSFIND, 175
    entrada em MDSCAL, 217
  de distâncias, 180, 346
    produzida por CONFIG, 180
  de estatísticas, 283
    produzidas por TABLES, 286
  de produtos
    cruzados, 207, 256, 365, 366, 398
    escalares, 180, 346, 359
  de relações, 195, 196, 261, 358, 404, 405
  de similaridades
    entrada em CLUSFIND, 175
    entrada em MDSCAL, 217
  de somas de quadrados, 207, 365, 366
  exportação
    (formato livre), 137
  importação
    (formato livre), 137
  inversa, 207, 366
  no setup de entrada, 22
  projeção, 325
  quadrada, 17
  retangular, 18
  vetor de médias e DP's, 18
mensagens de erro, 435
Minkowski (métrica-r), 215, 374

nome de variável, 15
normalização
  de configuração, 345, 371
  de matriz de relação, 261, 406

outliers
  definição, 229, 279
  detecção e eliminação, 228
  identificação e impressão, 276

padronização
  de medidas, 173, 337
  de variáveis, 428
palavras-chave
  para parâmetros comuns, 30
  regras de codificação, 29
  tipos, 28
parâmetros
  apresentação no Manual, 27
  comuns
    BADDATA, 30
    INFILE, 30
    MAXCASES, 30
    MDVALUES, 30
    OUTFILE, 30
    VARS, 31
    WEIGHT, 30
  declarações de parâmetro, 27
  localização, 27
  regras de codificação, 29
  tipos de palavras-chave, 28
  valores default, 27
partição ao redor de medoides, 174, 338, 340
partições binárias, 275, 413, 415, 416
Pearson (coeficiente de correlação r), 255, 397, 412
Phi (estatística), 308
plotando diagramas de dispersão, 269
ponderando dados, 30
preferência
  estrita, 262
  exemplo, 263
  fraca, 262
  tipos de, 262, 401

quantis, 191, 285, 353, 420

Recode
  acessando a facilidade Recode, 22
  constantes
    de caracteres, 35
    numéricas, 35
  declarações, 46
  elementos de linguagem, 35
  expressões, 36
    aritméticas, 36
    lógicas, 36

formato das declarações, 33
funções
aritméticas, 37
lógicas, 45
inicialização dos valores das variáveis, 34
linha de continuação, 33
manuseio de dados perdidos, 34
operadores
aritméticos, 36
lógicos, 36
relacionais, 36
operandos, 35
restrições, 54
teste, 35
variáveis V e R, 35
verificação de sintaxe, 92
Recode, declarações
BRANCH, 48
CARRY, 50
CONTINUE, 48
de designação, 46
DUMMY, 46
ENDFILE, 48
ERROR, 48
GO TO, 49
IF, 49
MDCODES, 51
NAME, 51
REJECT, 49
RELEASE, 49
RETURN, 49
SELECT, 47
Recode, funções aritméticas
ABS, 37
BRAC, 37
COMBINE, 38
COUNT, 39
LOG, 39
MAX, 40
MD1, MD2, 40
MEAN, 40
MIN, 40
NMISS, 41
NVALID, 41
RAND, 41
RECODE, 41
SELECT, 42
SQRT, 43
STD, 43
SUM, 43
TABLE, 43
TRUNC, 44
VAR, 44
Recode, funções lógicas
EOF, 45
INLIST, 45
MDATA, 45
recodificando dados, 31, 33, 59
exemplo, 33, 52, 60
salvando variáveis recodificadas, 165
registro
deleção de registro inválido, 122
duplicado detecção e deleção, 122
perdido detecção e padding, 122
registros-C, 15
listagem, 145
uso na validação de dados, 109
registros-T, 15
regressão, 205, 256, 269, 365, 398, 412
com intercepto zero, 370
com variáveis categóricas, 205, 211, 223
com variáveis dummy, 205, 211
linear múltipla, 205, 365
linhas de, 322
stepwise, 205, 369
stepwise descendente, 205, 370
resíduos, 369, 380, 415–417
produzidos por MCA, 223, 226
produzidos por REGRESSN, 206, 208
produzidos por SEARCH, 275, 276
ro de Spearman, 283, 422
rotação de configuração, 179, 345
rotação varimax
de fatores, 364
da configuração, 180, 346
de fatores, 196

séries temporais
análise, 329
transformação, 332
salvando variáveis recodificadas, 165
selecionando casos com filtro, 25
Spearman (ro de), 283, 422
Student (teste-t), 283, 426

título
declaração de controle, 27
localização, 27
regras de codificação, 27
tabelas
bivariadas, 283
multidimensionais, 307
univariadas, 283
taus de Kendall, 283, 308, 422
testando
declarações de controle de programa, 30
declarações de Recode, 35
teste
chi-quadrado, 283, 308, 420
D de Kolmogorov-Smirnov, 191, 194, 354
de Durbin-Watson, 207, 369
F de Fisher, 207, 225, 242, 367, 391
não-paramétrico
Fisher (exato), 283, 424
Mann-Whitney, 283, 425
Wilcoxon (signed ranks), 283, 425
t de Student, 283, 426
transformação
de configuração, 346

de séries temporais, 332
de configuração, 179
de dados, 59, 165

univariadas
estatísticas, 196, 207, 269, 283, 305, 306, 322, 333, 357, 411, 419
tabelas, 283, 307
apresentação gráfica, 308
produzidas por TABLES, 286

V de Cramer, 283, 308, 421
validação de dados, 57, 109
valores de dados não-numéricos, 13
detecção, 103
edição, 30, 103
valores próprios, 359, 360
valores randômicos
geração por Recode, 41
variáveis
categóricas na regressão, 205
dummy
criação com Recode, 46
usadas na regressão, 205
numéricas, 103
regras de codificação, 12
variável
agregada, 97, 98
alfabética, 13
ativa, 195, 295, 360, 427
correção, 129
decimal, 12
dummy, 46
número, 12, 15
número de referência, 15
nome, 15, 51
numérica, 12
edição, 14, 103, 105
regras de codificação, 12
passiva, 195, 295, 362, 427
principal, 195, 360
registro de descrição, 15
suplementar, 195, 362
tipo, 15
variância (análise de), 241
vetores próprios, 359

Wilcoxon (teste signed ranks), 283, 425
WinIDAMS
arquivos, 79
folders, 80
Interface do Usuário
customização do ambiente, 83